VAASAN AMMATTIKORKEAKOULU
VASA YRKESHÖGSKOLA
UNIVERSITY OF APPLIED SCIENCES

Kenth Joki

VAPAAN LÄHDEKOODIN VERKONHALLINTAPALVELIMEN TUOTTEISTUS

Tekniikka ja liikenne
2011

VAASAN AMMATTIKORKEAKOULU
Tietotekniikka

TIIVISTELMÄ

Tekijä	Kenth Joki
Opinnäytetyön nimi	Vapaan lähdekoodin verkonhallintapalvelimen tuotteistus
Vuosi	2011
Kieli	suomi
Sivumäärä	45
Ohjaaja	Antti Virtanen

Opinnäytetyön tavoitteena on toteuttaa, vapaan lähdekoodin lisenssin alainen, verkonhallintapalvelin Suupohjan Seutuverkko Oy:n valokuituverkon kytkinten toiminnallisen valvonnan edistämiseksi. Palvelimelle asetettiin yrityksen puolelta tietyt vaatimukset, jotka tulisi toteutua. Lopputuloksen tuli olla, toiminnallisten ominaisuuksien lisäksi, mahdollisimman käyttäjäystävällinen ja selkeä hallintaratkaisu.

Verkonhallintapalvelimen käyttöjärjestelmäksi valittiin Linux kernel-pohjainen Ubuntu ja hallintaohjelmistoksi Nagios Core, jonka ympärille ohjelmallinen kokonaisuus toteutettiin. Palvelin, SSH-palvelinohjelmisto asennettuna, liitettiin valokuituverkkoon yrityksen Kauhajoen toimitiloissa ja tälle määrättiin staattinen yhteys ulkoverkkoon. Asennus ja konfigurointi toteutettiin SSH-tunnelointia hyödyntäen. Työn kuvauksessa esitellään tehdyt ohjelmien asennukset kuten myös näiden toiminta ja tarkoitus. Palvelinohjelmiston vaatimat lisäkomponentit kuvataan ja lukijalle annetaan yleiskuva näiden teoreettisesta toiminnasta. Verkonhallintapalvelimen toiminnallisuudesta annetaan kattava esitelmä jossa selostetaan tällä toteutetut vaatimukset ja tämän mahdollinen laajennettavuus, niin sovellusten kuin hallittavan verkon suhteen.

Palvelinohjelmisto todetaan toimivaksi, joskin työlääksi ratkaisuksi. Yrityskäytössä, vaikkakin ohjelmiston hankintakustannusten suhteen ilmainen ja toimiva ratkaisu, vaatii suuren panostuksen ja paljon työtunteja yrityksen työntekijöiltä. Palvelinohjelmiston ylläpito ja tämän mukautettavuus yrityksen tarpeiden ja ilmeen mukaiseksi tekee Nagios-sovelluksella toteutetusta palvelimesta kuitenkin alustavan panostuksen arvoisen verkonhallintajärjestelmän, yrityksen resurssien tämän salliessa.

Avainsanat verkonhallinta, palvelin, hallinnointi, Nagios, Linux

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Tietotekniikka

ABSTRACT

Author	Kenth Joki
Title	Commercialization of an Open Source Network Management Server
University	University of applied sciences
Year	2011
Language	Finnish
Pages	45
Name of Supervisor	Antti Virtanen

The goal of this thesis is to implement an open source licensed network management server for the purpose of expanding the operational control of the switches in use in the fiber optic network owned by Suupohjan Seutuverkko Oy. A number of requirements to be realized were set by the firm. The outcome, in addition to the operational properties, should be as user-friendly and clear a management solution as possible.

For this network management server, the Linux kernel based Ubuntu was chosen as the operating system, and Nagios Core, around which the programmatic entity would be created, as the management software. The management server, with SSH-server software installed, was connected to the fiber optic network from the company premises in Kauhajoki, and was given a static connection to the outside network. Installation and configuration was then carried out remotely using SSH-tunneling. In the description of implementations are presented the installation of the software, as well as their function and purpose. The additional components required by the server software are described and the reader is given an overview of the theoretical functions of these. A comprehensive presentation of the functionality of the network management server is given, in which is explained the fulfilled requirements and the scalability of both the software and the monitored network.

The server software is found to be a functional, although laborious, solution. For use in a company, although free, regarding software acquirement, and a functional solution, it requires great effort and a lot of hours from the company employees. The upkeep and adaptability of the server software, regarding the company's needs and image, makes a server implemented using Nagios software a network management server worthwhile the initial investment, if feasible with regard to the company resourcing.

Keywords network, management, server, Nagios, Linux

SISÄLLYS

KUVA- JA TAULUKKOLUETTELO

LYHENTEET

CGI = Common Gateway Interface

GPL = General Public License

HTML = Hypertext Markup Language

IP = Internet Protocol

LTS = Long Term Support

MIB = Management Information Bases

OID = Object Identifiers

OSI = Open Source Iniative

PHP = Hypertext Preprocessor

RAID = Redundant Array of Independent Disks

RAM = Random Access Memory

RRA = Round Robin Archives

RRD = Round Robin Database

SLA = Service Level Agreement

SNMP = Simple Network Management Protocol

SSH = Secure Shell

TCP = Transmission Control Protocol

UDP = User Datagram Protocol

WWW = World Wide Web

1 JOHDANTO

Suupohjan Seutuverkko Oy:n alati laajeneva valokuituverkko ja asiakaskanta johtaa yhä useamman verkkoon asennettavan laitteen ja näiden linkkien valvomista. Asiakkaiden yhteyksien jatkuva toimivuus edistää yrityksen kannattavuutta niin resurssien käytön kuin maineen suhteen. Mahdollisten huoltotöiden suorittaminen tulee olla yksinkertainen operaatio. Hallinnoitavassa verkossa esiintyvän virheen havainnointi, paikannus ja tämän vaatimat toimenpiteet tulee mieluiten sujua lyhyessä ajassa, asiakkaan huomaamatta ja aiheuttaen mahdollisimman vähän kustannettavien resurssien käyttöä. Tämä vaatii jokaisen ylläpidettävän laitteen ja yhteyden jatkuvaa, valpasta seurantaa. Ainoastaan automatisoitu seuranta on mahdollista näin laajassa verkossa. Tärkeintä on saada tieto virheestä heti tämän esiinnyttyä. Tieto on saatava viestinä oikealle henkilölle tavalla, jolla tämä välittömästi havaitaan. Viestin täytyy sisältää virheen tiedot kuten tyypin ja sijainnin verkossa. Tämän viestin aikaansaamiseksi on Suupohjan seutuverkolla oltava valokuituverkon hallintajärjestelmä.

Opinnäytetyön päätavoitteena on toteuttaa kustannustehokas ja täten ilmaiseen lähdekoodiin perustuva, nykyistä järjestelmää vastaava tai tätä parempi, verkonhallintapalvelinratkaisu. Käyttäjäystävällisyys tulee huomioida niin käyttöjärjestelmän kuin ohjelmiston suhteen. Täten tämän tulee pohjautua graafisella käyttöliittymällä olevaan käyttöjärjestelmään. Myös hallintaa mahdollistavan ohjelmiston tulee olla mahdollisimman selkeä ja kuvaava. Lopullisesta hallintajärjestelmästä tulee laatia käyttöä opastava yleisohje yrityksen käyttöön. Opinnäytetyötyö, joka tulee osittain suorittaa Kauhajoella, Suupohjan Seutuverkko Oy:n tiloissa, sisältää myös yrityksen asiakastietokantojen järjestelyä xml-taulukoissa. Kyseisiä taulukoita yritys käyttää verkossa olevien kytkinten porttien määrittelyyn.

2 TARVEANALYYSI

Lopputyö perustuu keväästä 2009 Vaasan ammattikorkeakoulun tietoliikennetekniikan suuntautumisvaihtoehdon kolmannen moduulin laboratoriotyöhön. Tässä, yhteistyössä Suupohjan Seutuverkko Oy:n henkilöstön kanssa, opiskelijoiden tuli tutkia ja esitellä vaatimuksiltaan tämän lopputyön tavoitteita vastaavia verkonhallintapalvelinohjelmistoja. Suupohjan Seutuverkko Oy:n nykyiset verkonhallintamenetelmät ovat työläitä kytkinten konfiguroinnin ollessa ylläpitäjän käsin syötettävissä, jokaiseen käytössä olevaan kytkimeen. Yrityksen kytkinten määrän lähentyessä sataa yksikköä, jokaisen näiden käytössä olevan portin asetusten määrääminen vie huomattavia resursseja tekijöiden ja ajan suhteen. Tämän osittainen automatisointi on yhtiöllä työn alla. Tähän tarkoitukseen käytettävää Netadmin-ohjelmistoa hyödynnettäessä xml-taulukkoihin asetettujen kytkinkohtaisten tietojen syöttö tapahtuu ajamalla taulukon sisältö etäältä kytkimen käyttöjärjestelmään. Näin asiakaskohtaisesti käytössä olevat portit määritellään jokaiselle kytkimelle huomattavilla säästöillä resursseissa.

Valokuituverkon ja tässä tarjotun palvelun ylläpitoa edistävää ja mahdollisimman edullista ratkaisua haetaan avoimen lähdekoodin pohjalta luoduista olemassa olevista hallintaohjelmista. Avoimen lähdekoodin ohjelmisto on vapaasti käytettävissä ja muokattavissa myös yrityskäytössä, Open Source Initiativen (OSI:n) määräämien vaatimusten mukaisesti /1/. Nämä ovat valtaosaltaan Linux-käyttöjärjestelmä-pohjaisia mutta ovat riippuvaisia jokainen omastaan käyttöympäristöstä. Eli jokaisella ratkaisulla, Linux-jakeluvalinnasta riippuen, on omat vaatimukset siihen mitä käyttöjärjestelmään tulee tämän lisäksi asentaa ja konfiguroida.

2.1 Vaatimukset

Ratkaisun tulisi olla täysin vapaaseen tuotekoodiin perustuva ratkaisu niin käyttöympäristön kuin ohjelmiston suhteen. Palvelimella tulisi olla graafinen käyttöliittymä, jolla tämä esittää verkkoliikenteen kuormituksia ja muita tilastoja kuvaajien muodossa. Tilastojen tulee olla mahdollisimman tarkkoja ja näiden

historia tulee näkyä mahdollisimman pitkältä ajalta. Graafisen käyttöliittymän tärkeys perustuu käyttäjäystävällisyyden tärkeyteen. Palvelimen tulisi kyetä seurata verkossa olevien kytkinten linkkitilaa kuten myös informoida ylläpitäjää mahdollisista katkoista ja virhetilanteista. Näistä ohjelmiston tulisi lähettää hälytyksiä sähköposteina ja tekstiviesteinä määrättyihin osoitteisiin ja puhelinnumeroihin. Linkeistä tulee myös selvittää näiden SLA-arvo (Service Level Agreement), eli prosentuaalinen suhde linkin toimivuudesta (uptime/ downtime). Tätä arvoa käytetään yrityksen palvelun laadun kuvaamiseen. Kytkinten lokitietoja tulee kerätä määrättyihin sijainteihin verkossa. Hallintaohjelmiston tulee kyetä monitoroida vähintään kuuttasataa (600) porttia seitsemässäkymmenessä (70) laitteessa. Kyseiselle palvelimelle tulee arvioida vaatimukset raudan suhteen käyttötarkoitukseen sopivaksi ja käyttöjärjestelmää ja ohjelmistoa tukevaksi. Tässä tulee huomioida emolevyn, keskusyksikön, näytönohjaimen, käyttömuistin (RAM) ja kiintolevytilan kuten myös tähän käytetyn varmuuskopiointijärjestelmän.

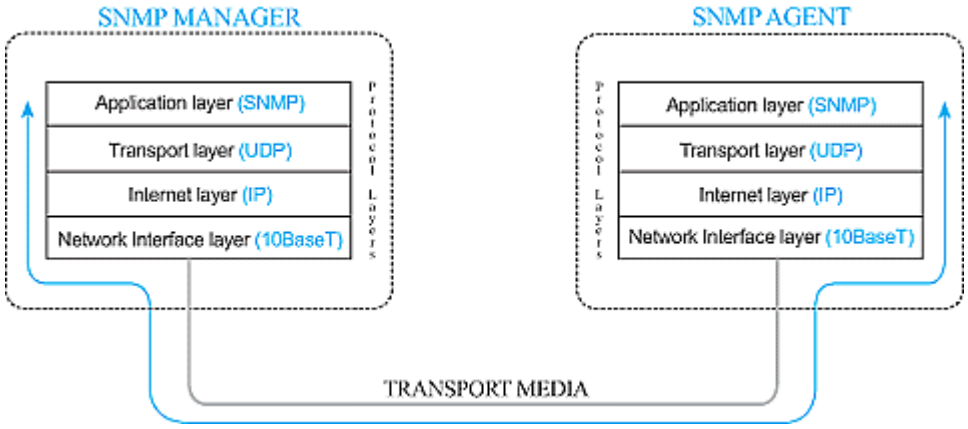
2.2 Tavoitteet

Opinnäytetyö tavoitteena on suunnitella ja tuotteistaa Suupohjan Seutuverkko Oy:n valokuituverkossa toimiva avoimen lähdekoodin verkonhallintapalvelin ja tätä myöten saada henkilökohtaista kokemusta ja osaamista valokuituverkon toiminnasta ja vastaavassa verkossa esiintyvien laitteiden tarpeista ja näiden tarpeiden tyydyttämisestä erin tietoteknisin ratkaisuin. Valmiin verkonhallintapalvelimen tulisi olla mahdollisimman käyttäjäystävällinen, niin tulkittavuuden ja käytettävyyden kuin asetusten ja lisäparametrien määrittelyn suhteen. Lopputuloksen tulisi täyttää kaikki Suupohjan Seutuverkko Oy:n hallintapalvelimelle asettamat vaatimukset (katso kappale 2.1).

3 OHEISOHJELMISTON TEOREETTINEN KUVAUS

3.1 SNMP

SNMP, lyhenne sanoista Simple Network Management Protocol, on OSI-mallin sovelluskerroksella toimiva verkonhallintaprotokolla, jolla verkossa toimivat laitteet kuten esimerkiksi palvelimet, reitittimet ja kytkimet, saattavat välittää hallinnointi-informaatiota keskenään ja ulkoisiin verkkoihin. SNMP-paketit, pakattuina UDP-paketteihin, kulkevat IP-kerroksella fyysisen linkin kautta SNMP-hallintaohjelmiston ja hallinta-agentin välittämänä datana (kuva 1.)



Kuva 1. SNMP-datasiirto /2/.

Protokollasta on versiot SNMPv1, SNMPv2c ja SNMPv3. Versioiden erot esiintyvät näiden tiedonkeruun määritelmissä, tyypeissä kuten myös tietoturvamääritelmissä. SNMPv3 vaatii useamman parametrin hakujen suorittamiseen kun aiemman versiot, kuten määritellyn käyttäjän ja salasanan. SNMP-agentit, eli ohjelmalliset hallinnoitavilla laitteilla olevat SNMP-komponentit, keräävät ja välittävät, oikeutetun hallintalaitteen suorittamien SNMP-kyselyiden pohjalta, vastaukset kyseisestä laitteesta. /2/

Laitekohtaiset hallinnoitavat ominaisuudet ovat lueteltuja laitteiden MIBeissä (MIB, Management Information Bases). MIBit ovat ANS.1 standardien mukaisesti koottuja tietokantoja, jotka sisältävät laitteelle luodut OID:t (Object Identifiers). Jokainen OID määrää datan, jonka laitteesta voi lukea tai tälle

kirjoittaa. OID:n rakenne on hierarkkinen nimiavaruus, jonka jokainen kerros on yksinkertaistettu numerolla tai numerosarjalla Esimerkiksi OID "ifUpTime", joka ilmoittaa laitteen päälläoloajan, on nimiavaruutena "iso.org.dod.internet.mgmt.mib-2.system.ifUpTime" ja yksinkertaistettuna sarjana {0.1.3.6.1.2.1.2.1.1.3}. OID:t saa myös luettua tämän nimellä, jos tunnettu, esimerkiksi system.sysContact.0. /3/

Hallinnoitavilla laitteilla täytyy olla SNMP asennettuna ja konfiguroituna. Laitteilla täytyy olla määriteltynä sama "community", eli SNMP-ryhmä, johon laitteet kuuluvat. Jos SNMP-ryhmä ei ole konfiguroituna ohjelmistoon, kyseinen nimi täytyy määritellä kyselyn suorituksessa ja täten toimii myös varmenteena. Hallintalaitteistolla täytyy myös olla oikeudet kirjata komentoja hallittavalla laitteistolla olevalle agentille. Oikeudet voidaan määritellä joko luku- ja kirjoitusoikeudeksi (read-write), tai pelkästään lukuoikeudeksi (read-only). Hallintalaitteesta saattaa hakea agenteista tietoa useammalla komennolla. Nämä ovat get, get-next ja get-bulk, kuten myös näistä ohjelmallisesti totutetut laajennetut pyynnöt kuten snmpwalk, joka hyödyntää getnext komentoa MIB-taulukon kaikkien OID:n haussa. Seuraavana esimerkki toteutettavasta hallintapalvelimen hausta hallinnoitavan laitteen agentille:

snmpget -v1 -c public 192.168.0.1 system.sysContact.0

Hallintajärjestelmä luo halutun OID:n tunnisteesta get-paketin, joka lähetetään, esimerkiksi reitittimen, hallinta-agentille. Agentti selvittää OID:n MIB-tietokannasta. OID:n löydyttyä, agentin luoma get-response-paketti OID:n nykyisestä arvosta lähetetään hallintajärjestelmälle. Esimerkissä on määritelty komento get, versiolla SNMPv1 ja tämän communityksi "public". Haku OID:n "system.sysContact.0" datasta, eli haku laitteen hallinnoijan yhteystiedoista, kohdistuu laitteeseen jolla IP-osoite 192.168.0.1. Kyseiseen hakuun agentti vastaa esimerkiksi seuraavanlaisella vastauksella, sisältäen kysellyn merkkijonon (get-response):

system.sysContact.0 = STRING: nagios@localhost

SNMP mahdollistaa myös asetusten määrittelemisen hallittavassa laitteistossa, tämän agentin kautta OID:lle määrättävillä arvoilla. Tämä suoritetaan komennolla set. Esimerkiksi tietotyypin INTEGER ("i"), STRING ("s"), HEX STRING ("x"), DECIMAL STRING ("d") tai IP-address ("a") mukaisesti asetettava data määrätään OID:lle seuraavasti:

snmpset -v1 -c public 192.168.0.1 system.sysContact.0 s nagadmin@localhost

Komennolla asetetaan OID:lle "system.sysContact.0" STRING ("s") -tyyppinen merkkijono "nagadmin@localhost". /3/

Hallinta-agentin saa myös automaattisesti lähettämään tietoa, parametrien mukaisesti, tärkeiksi määritellyistä tapahtumista. Nämä trap-makrot ovat määriteltävissä agentin SNMP-version mukaisen MIBin tietoihin. Määriteltynä ovat OID:n arvot, jotka aiheuttavat ilmoituksen ja ilmoituksen sisältö ja rakenne. Agentin lähettämä paketti tulisi sisältää OID:n, sen hetkisen arvon lisäksi myös OID:n tunnisteen, vian paikantamiseksi. /2/

3.2 PHP

PHP, rekursiivinen akronyymi sanoille PHP: Hypertext Preprocessor, on skriptikieli, joka soveltuu erityisen hyvin dynaamisten web-sivujen luontiin ja käsittelyyn. PHP-skripti on HTML-tiedostoon sisäistetty ohjelmallinen kieli, joka ajetaan palvelimen puolella asiakkaan selaimen tätä pyytäessä. HTML-tiedostossa esiintyy prosessoinnin aloitus- ja lopetusilmaisimet, joiden välissä sijaitsevan ajettavan koodin palvelimen PHP-komponentti tulkitsee ja käsittelee ja palvelin lähettää täysin HTML-pohjaisen sivun takaisin asiakkaalle, tämän näkemättä suoritettavan koodin sisältöä. Seuraavana esimerkki palvelimella olevassa HTML-tiedostossa esiintyvä PHP-skripti:

<?php echo '<p>PHP-esimerkki</p>'; ?>

Kyseinen skripti näkyy, palvelimen PHP-tulkin kautta, asiakkaalle lähetettävässä HTML-tiedostossa pelkkänä HTML-koodina seuraavasti:

** PHP-esimerkki **

Eli asiakkaan selaimesta saattaa lukee tekstin ”PHP-esimerkki”. Kyseinen toimintatapa luo tietoturvaa palvelimen ja asiakkaan välille, ohjelmallisten parametrien pysyessä web-palvelimella. /4/

3.3 RRDtool

RRDtool, eli Round Robin Database tool, on loki- ja analysointiohjelmisto, jonka avulla kerätään informaatiota tiedonlähteistä ja nämä tallennettuna tietokantaan mahdollistetaan tiedon myöhempää hyödyntämistä verrannolliseen analysointiin, esimerkiksi kuvaajien muodossa. Hallinnoitavalta laitteelta tai palvelulta, esimerkiksi minuutin välein, vuoden ajan kerätty tieto aiheuttaa huomattavaa tarvetta tallennustilalle ja kuormittaa analysointiin käytettävän datan läpikäynnissä järjestelmää. RRDtool antaa määritellä parametrit datan keruuseen ja tallentamiseen. Määrättävissä on esimerkiksi tietokantaan tallennettujen arvojen määrä ja näille käytettävät raja-arvot. Määrän rajaus, esimerkiksi 500 arvoa tietokantaa kohden, rajaa mitattujen arvojen määrän tietokannassa, ja kun tämä määrä tulee täyteen, arvot arkistoidaan RRA-tiedostoihin (Round Robin Archives). Kyseiseen arkistointitiedostoon otetaan näytteitä kerätyistä arvoista ja tallennetaan ainoastaan oleelliset muuttuvat arvot ja muutosaika. Näin jatkuvat, pysyvät, arvot voidaan hävittää, mutta kokonaiskuva arvoista ja näiden muutoksista saadaan arkistoitua. Ääriraja-arvot ovat käytännöllisiä tilapäisen tiedonkeruun katkon tai häiriön sattuessa, tämän aiheuttaen tuntemattoman tai vääristävän arvon, jonka saa näin määrätä hylättäväksi. RRDtool sisältää myös algoritmisen poikkeavuustarkistuksen tulevalle mitattavalle arvolle. Suuruuden, lineaarisen trendin ja tyypillisen jaksopoikkeaman huomioon ottaen tämä vertaa arvioidun arvon ja mitatun arvon eroavaisuuden ja jatkuvan arvojen poikkeaman täyttäessä laskennalliset kriteerit tämä kirjaa arvon erilliseen FAILURES RRA-arkistoon, jota käyttäjä saattaa hyödyntää hälytysilmoituksien luontiin. /5/

3.4 Apache2 ja CGI

Apache Software Foundationin ylläpitämä Apache2 Web Server on web-palvelinohjelmisto, jolla kyseisen ohjelmiston omaava palvelin, jakaa HTTP-, HTTPS- ja FTP-protokollapohjaista sisältöä asiakkaan Internet-selaimen

välityksellä. Asiakkaan tehdessä pyynnön kyseisen palvelimen kiinteälle DNS-rekisteröidylle IP-osoitteelle, TCP/IP-verkon välityksellä, palvelin lähettää vastaavan pyydetyn tiedoston datan asiakkaalle joko sellaisenaan tai dynaamisesti, ohjelmallisesti muunneltuna. Useimmiten kyseinen data on HTML-tiedosto, jonka yhteydessä palvelin lähettää myös sivustoon sisällytetyt mediatiedostot. Apache2 ohjelmisto on Apache-lisenssin, versio 2.0 alainen, jolloin tämä on vapaata lähdekoodia ja on käytettävissä myös yrityskäytössä. Yrityksille myös tarjotaan maksullisia toteutuksia ja tukea. Ohjelmisto tukee lukuisia ohjelmointi- ja skriptikieliä, kuten PHP, Perl ja Python, kuten myös eri salausmenetelmiä kuten SSL. Web-palvelinohjelmisto on tehokas myös valtavan rasituksen alla, järkevästi optimoidun rakenteen ansiosta. Ohjelmisto on määrätty varautumaan kuormaan jo käynnistyessä ja on näin valmiustilassa, jos ohjelmisto saa kuormittavan määrän kyselyitä. /6/

CGI (Common Gateway Interface) on sovelluskomponentti, jolla Apache2 web-palvelimelle saadaan lisättyä dynaamista sisältöä erillisistä ohjelmista. Vastaavasti asiakkaan selaimesta saadaan tieto CGI-skriptien kautta palvelimella suoriutuville ohjelmille. CGI-skripteillä muuttuva tieto sisällytetään HTTP-sivuston mukaisen otsikoinnin avulla asiakkaan selaimelle. /6/

4 PALVELIMEN SUUNNITTELU

4.1 Yleistä

Suunnittelu toteutetaan verkonhallintapalvelimelle asetettujen vaatimusten perusteella, alkaen näiden vaatimusten luomasta tarpeesta laitteistosta saatavalle toiminnalliselle suorituskyvylle. Laitteisto voi itsenään luoda tarpeita ohjelmiston kannalta, joiden täytyy myös olla yhteneväisiä palvelimen vaatimusten suhteen, joten suunnitteluvaiheessa täytyy huomioida kokonaisuus ja tarkoituksen-mukaisuus.

4.2 Laitteet ja komponentit

Suupohjan Seutuverkko Oy:n verkonhallintapalvelimen laitteistoon tai näiden komponentteihin ei kohdistu mainittavaa ulkopuolista rasitusta. Laitteisto tulee sijaitsemaan sisätiloissa, ilmastoidussa huoneessa joten lämpötila, kosteus, runsas pöly tai muu vastaava tekijä ei heikennä koneen suorituskykyä tai lyhennä tämän toimintaikää. Näin nykystandardien mukainen peruskäyttöön kelpaava pöytätietokone on pätevä. Käyttöjärjestelmän ja ohjelmiston asettamat vaatimukset palvelimelle kohdistuvat keskusyksikköön, ja täten emolevyyn ja muuhun laitteistoon. Käyttöjärjestelmän ja siihen asennetun ohjelmiston tarpeellisen suorituskyvyn varmistamiseksi, näiden tulee olla 64-bittisiä. Kuudenkymmenenneljän bitin keskusyksikkö ja ohjelmisto, yhdessä 4 gigatavun käyttömuistin kanssa nopeuttaa palvelimen prosessointiaikaa ja edesauttaa kokonaisuuden toimivuutta mahdollisen raskaamman kuorman alla. Vastaava palvelinrakenne luo myös yhteensopivuutta kaikissa ohjelmallisissa ratkaisuissa, myös vapaan lähdekoodin alaisen palvelinohjelmiston usein nykyään ollessa 64-bittisiä.

Emolevyyn kohdistuvat vaatimukset, keskusyksikön kannan yhteensopivuuden lisäksi, on tarve piirille joka mahdollistaa RAID1-, eli peilaus-varmuuskopioinnin. Kyseisellä varmennuksella palvelimen kiintolevyllä oleva tieto kopioituu yhdelle tai useammalle toiselle kiintolevylle. Näin tieto on palautettavissa yhden

kiintolevyn hajotessa. Jos kyseistä piiriä ei löydy, tulee tälle asentaa vastaava lisäkomponentti.

Kiintolevyjen tarpeellinen minimitilavuus on arviolta noin 10 gigatavua. Tämän tulisi kattaa käyttöjärjestelmän ja ohjelmiston asennuksen kuten myös palvelimen hallittavista laitteista keräämä tieto. Kyseisen kerätyn tiedon tilavaatimus on useimmilla ohjelmistoratkaisuilla käyttäjän määrättävissä, lokitiedostojen automaattisen kierron kautta, määräämällä tiedostolle tietty sallittu suuruus tai ikä, ennen kuin alkuperäinen vanhin tieto korvataan uudella.

Muiden komponenttien kuten näytönohjaimen, verkkokortin, CD- tai DVD-aseman ja palvelimen virtalähteen suhteen, laitteiston ja ohjelmiston kanssa yhteensopivat, yleispätevät komponentit riittävät tarpeisiin. Sujuvan liikennöinnin varmistamiseksi verkossa, verkkokortin nopeudeksi suositellaan vähintään 100 Mbit/s vastaanotolle (Rx) ja lähetykselle (Tx).

4.3 Käyttöjärjestelmä

Alusta alkaen verkonhallintapalvelimen käyttöjärjestelmäksi todettiin tulevan joku julkaistuista Linux-pohjaisista käyttöjärjestelmistä, vaatimusmäärittelyn mukaisesti. Tarveanalyysissä mainitun laboratoriotyön aikana kyseinen aihetta käsittelevä ryhmä totesi graafisen käyttöliittymän omaavan Ubuntu 8.04 LTS Linux-käyttöjärjestelmän, nimetty ”Hardy Heron”, julkaistu huhtikuussa 2008, parhaaksi vaihtoehdoksi käytettävyyden ja graafisten ominaisuuksiensa ansiosta /7/. Kaikille LTS, ”Long Term Support” käsitteen omaaville Ubuntu julkaisuille, joissa graafinen Gnome-työpöytäjärjestelmä, luvataan kehittäjien puolelta kolmesta viiteen vuotta tietoturva- ja päivitystukea. Opinnäytetyön toteuttamisvaiheessa, vastaava päivitetty versio käyttöjärjestelmästä oli Ubuntu 10.04 LTS, ”Lycid Lynx”, julkaistu Huhtikuussa 2010. Kyseiselle julkaisulle on luvattu tukea huhtikuuhun 2013. /8/

4.4 Ohjelmisto

Yrityksen sisäiseen käyttöön tuleva verkonhallintapalvelin tulee olla vahvan tietoturvan alainen. Käyttöjärjestelmä mahdollistaa tämän sisäisen palomuurin

Iptables avulla. Kyseinen sovellus asentuu käyttöjärjestelmän mukana ja tälle tulee ainoastaan määrätä sääntöjä sallituista ja vastaavasti kielletyistä yhteyksistä yksittäisille osoitteille tai osoiteavaruuksille. Säännöt voidaan myös määrätä tietoliikenneprotokollakohtaisesti, jolloin tietyt avoimemmat ja näin riskialttiimpien protokollien käyttöä voidaan rajoittaa tai täysin kieltää. Sovellus tarkistaa lähtevät ja saapuvat paketit ja soveltaa määrättyjä sääntöjä paketin kehysten sisältämän tiedon arviointiin. Yksinkertaisin ja kyseiseen käyttöön parhaiten soveltuva sääntö on määrätä pakettien salliminen ainoastaan tarkoituksen mukaisista ulkoisista osoitteista ja kieltää muu, kuin näistä osoitteista perustetut yhteydet ja näiden luoma liikenne.

Hallintaohjelmistolle oli useampi pätevä vaihtoehto, joista Nagios ja Zenoss olivat todennäköisimmät vaihtoehdot tarjolla olevista vapaan lähdekoodin sovelluksista. Aiemmin, kappaleessa 2, sivulla 8 mainitun laboratoriotyön lopputuloksen mukaisesti, Zenoss on pätevä vaihtoehto, mutta ainoastaan tämän maksullinen versio kykenee hallinnoimaan tarvittavaa määrää laitteita. Myös tälle käyttöön otettavat, yhteisön toteuttamat, ominaisuudet ovat ensivaikutelman mukaan alkeellisemmat kuin esimerkiksi Nagioksella. Näiden dokumentaatio myös vaikuttaa puutteelliselta.

Nagios Core, Nagios-sovelluksen ilmainen jakeluversio, oli myös kyseisen laboratoriotyön tarkastelemana, mutta tällöin ryhmän työntekijät totesivat kyseisen sovelluksen liiankin vaativaksi monimutkaisen asennus- ja konfigurointiprosessin ansiosta, tällöin ryhmällä olleen tiukan aikarajoituksenkin takia. Suupohjan Seutuverkko Oy:n verkonhallintajärjestelmälle Nagios Core vaikutti kuitenkin varsin varteen otettavalta vaihtoehdolta lukuisien, hyvin dokumentoitujen, yhteisön luomien ratkaisujen ansiosta. Kummankin mainitun verkonhallintapalvelinohjelmiston tarjoama käyttökokemus perustui pitkälti näille luotuun, ja saatavilla olevaan dokumentaation, kuten myös yhteisön tarjoamaan tukeen. Nagioksen kattavan dokumentaation ja yhteisön suuruuden varjossa Zenoss ei ollut tarkemman tutkimuksen alainen ja opinnäytetyö suuntautui Nagiokseen pohjautuvaan palvelinohjelmistoon.

5 PALVELIMEN TOTEUTUS

Nagios, alun perin Linuxille luotu nimellä NetSaint, on Ethan Gelstadin luoma ja hänen kehitysryhmänsä ja nykyisen valtavan yhteisön ylläpitämä verkonhallintasovellus /9/. Nagios Core on GNU GPL -lisenssin (versio 2), eli vapaan lähdekoodin lisenssin alainen ja täten ilmainen sovellus, niin yksityiseen kuten myös kaupalliseen käyttöön /10/. Nagios Core mahdollistaa, vaikkakin kovalla vaivalla ja vain syvällisellä perehtymisellä, kattavan verkonhallintaohjelmiston toteuttamisen, ja on hyvin mukautettavissa jokaisen tietoliikenneverkon tarpeiden mukaan. Kyseisellä ohjelmistolla jokainen Suupohjan Seutuverkko Oy:n määräämistä vaatimuksista (katso kappale 2.1) on mahdollista toteuttaa, ja tämä tuli myös opinnäytetyöntekijän tavoitteeksi. Varsinainen ohjelmisto ei itsessään sisällä valmiita hallintatyökaluja, vaan nämä on asennettava ja konfiguroitava erikseen Coren implementoinnin jälkeen, käyttäen itse ohjelmoituja tai saatavilla olevia komponentteja (plugins). Nagios Core on ainoastaan laajasti mukautettavissa oleva runko hallintajärjestelmän luonnille.

Nagios Corella saattaa valvoa koko tämän saatavilla olevan tietoliikenneverkon laitteistoa, järjestelmiä, sovelluksia ja näiden tarjoamia palveluita. Jokaiselle valvottavalle laitteelle on määrättävä tiedot kuten nimi, verkko-osoite ja laitteistotyyppi Nagios Core:n konfigurointitiedostoihin, ja vastaavasti näiden laitteiden alaisille valvottaville palveluille määrätään tiedot kuten palvelun nimi, valvontaparametrit ja palvelun laadulle oleelliset kriittiset hälytysrajat. Opinnäytetyöhön käytettiin ainoastaan Nagios Coren valmista, viimeisintä tarjolla olevaa julkaisua ja tämän laajan yhteisön tarjoamia lisäkomponentteja. Omien komponenttien luonnin oletettiin alustavasti olevan turhaa, lukuisien tarpeenmukaisien ratkaisujen olemassaolon ansiosta.

5.1 Riippuvuudet

Nagios Core vaatii useamman ohjelmistokomponentin tietojen keruuseen ja kerätyn tiedon käsittelyyn. Näistä moni tulee asentaa ennen varsinaisen Coren asennusta, sovelluksen dokumentaation mukaisesti. Verkonhallintapalvelimen

käyttöliittymä on HTTP-pohjainen, eli sovelluksen käyttäjän liittymä on selaimessa, WWW-osoitteen pohjalta, avattava sivusto. Tämän palvelun mahdollistaa Apache 2 -ohjelmistokomponentti.

Libgd2-paketti on yksi monista Nagios Coren toimintaan tarvittavista komponenteista. Kyseinen grafiikka-kirjastopaketti mahdollistaa koodin pohjalta tapahtuvan dynaamisen kuvaajien ja muun grafiikan piirron. Kyseinen kirjasto tulee Coren käytössä tarpeen selaimeen esitettävien, toimintaa kuvaavien, WWW-sivujen luonnissa. /11/

PHP 5 -ohjelmisto, luo PHP, hypertext preprocessor, -ohjelmointikielellä sillan web-palvelimen ja Nagios Coren välille, jolla Nagioksen PHP-data tulkitaan web-palvelimelle esitettäväksi, dynaamisina web-sivuina. /4/

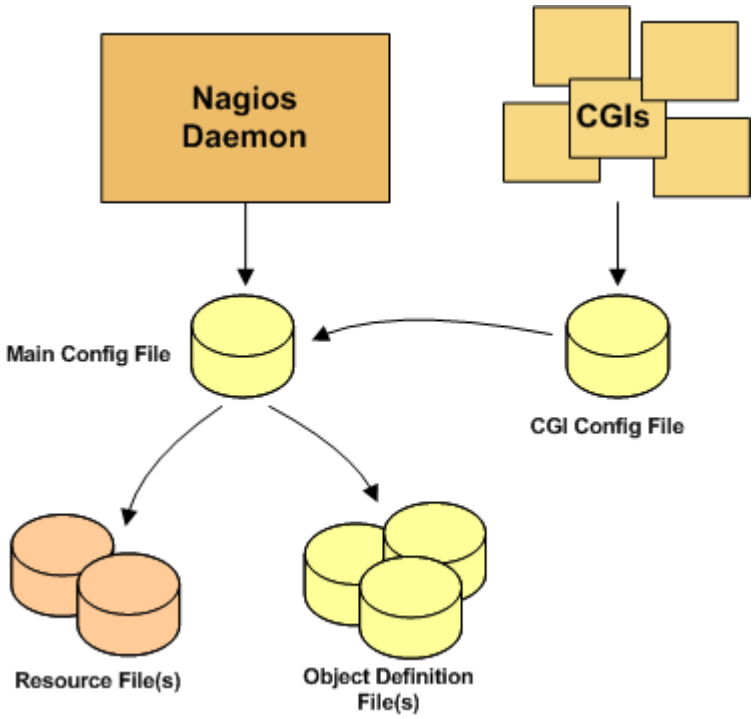
Net-SNMP on Nagioksen tarvitsema ohjelmistopaketti, jolla implementoidaan SNMP versiot v1, v2c ja v3, protokollilla IPv4 ja IPv6. Paketti sisältää komentokehoteohjelmiston tiedonkeruukomennoille snmpget, snmpgetnext, snmpwalk, snmptable ja snmpdelta, tiedonsyöttökomennoille snmpset. Paketti sisältää myös SNMP-ilmoitusten vastaanoton mahdollistavan snmptrapd-ohjelmiston kuten myös graafisen MIB-tietokannan selainohjelmiston tkmib. Paketissa on myös SNMP-hallinta-agentti, verkonhallintapalvelimelle suoritettavalle seurannalle. /15/

Palvelimelle asennettava RRD-tietokanta (Round Robin Database) mahdollistaa historiatiedon palveluista, esimerkiksi myöhemmin selitettyjen kuvaajien muodossa, ja tietojen takautuvan vertaamisen palvelun laadun selvittämiseksi. Kyseinen palvelun laadun selvittäminen hyvin oleellinen osa kattavaa verkonhallintaohjelmistoa. Nagios Coren keräämä tieto säilyy RRD-tietokannassa, josta määrättyjen ajettavien komentojen kautta tieto saadaan käyttäjän nähtäväksi. Näitä, hallittavista palveluista kertovia arvoja, takautuvasti tallentamalla palveluiden laadulliset raja-arvot saadaan myös dynaamisesti ohjelmiston tulkittavaksi, eli suuret tilapäiset heitot arvoissa ei välttämättä aiheuta hälytystilaa, vaan ainoastaan pysyvämmät erot arvoissa saa ohjelmiston hälyttämään määrättyä käyttäjää esimerkiksi sähköpostilla tai tekstiviestillä.

Etähallinnalle on vastaavan verkonhallintapalvelimen kannalta oleellinen tarve. Tämä on toteutettu SSH-tunneloinnilla ohjelmalla OpenSSH. SSH, akronyymi sanoille Secure SHell, on tietoturvan kannalta järkevä ratkaisu, siirrettävän datan salauksen ja yhteyksien autentikointiominaisuuksien ansiosta.

5.2 Ominaisuudet

Nagios Coren toiminta perustuu muiden ohjelmistojen tapaan usean komponentin yhteisestä toimivasta kokonaisuudesta (kuva 2.) Nagios Core daemon, eli varsinainen ohjelmallinen ydinkomponentti, toimii kaikkien muiden komponenttien pohjalta. Tätä toimivaa kokonaisuutta ohjataan useasta konfigurointitiedostoista. Tärkein näistä on nagios.cfg-tiedosto. Kyseistä tiedostoa lukee Nagios daemon ja Nagios CGI:t (Common Gateway Interface). CGI-skriptikieli toimii tiedonvälittäjäkomponenttina Nagios Coren ja Apache web-palvelimen tarjoaman selainpohjaisen käyttöliittymän välillä. Resurssitiedostot, Resource files, ovat erillisiä tiedostoja jotka määritellään Nagios Coren käyttöön tämän konfigurointitiedostoista. Kyseiset resurssit ovat ainoastaan daemonin luettavia ja eivät välity CGI:n kautta käyttöliittymälle, joten resursseihin on hyvä kirjata arempi tieto kuten SNMP-tietoturvamääritelmät ja salasanat. Laitteiden ja palveluiden hallintatoimenpiteet ja määritelmät kyseisistä hallittavista objekteista, määrätään pääosin Object Definition -tiedostoihin, joista pääkonfigurointitiedoston määritelmien pohjalta, Nagios daemon nämä käsittelee. CGI-konfigurointitiedostoon määrätään CGI-skriptien vuorovaikutus web-palvelimen kanssa. /12/



Kuva 2. Nagios Coren rakenne /13/.

5.2.1 Käyttäjähallinta

Nagios Core käyttöliittymä on selainpohjainen ja tälle kirjaudutaan kyseisen palvelimen ja sivustolle määrätyn osoitteen mukaisesti (kuva 3.) Käyttäjät määrätään joko Nagios Coren asennusta suorittaessa, tai jälkikäteen komentokehotteeseen syötetyllä komennolla, esimerkiksi seuraavasti:

htpasswd /usr/local/nagios/etc/htpasswd.users newuser1

Kyseinen komennonanto luo Nagios Coren käyttöliittymälle käyttäjän "newuser1" ja komentokehotteeseen tulee pyyntö antaa käyttäjälle salasana. Käyttäjien salasanat muutetaan vastaavalla komennolla, jossa käyttäjän nimeksi määrätään kyseinen jo olemassa oleva käyttäjä. Näille luoduille käyttäjille määrätään oikeudet, Nagios Core -käyttöliittymällä suoritettaville toimenpiteille, cgi.cfg-tiedostossa.

#cgi.cfg

use_authentication=1
authorized_for_system_information=nagiosadmin,user1,user2

Kuva 3. Kirjautuminen Nagios Core -käyttöliittymälle.

Itse palvelimen käyttöjärjestelmälle ja sovelluskomponenteille, kuten Nagios Coren cfg-tiedostoille, etähallintaa suorittavat käyttäjät määrätään Linux-käyttöjärjestelmälle ja komennot ovat näitä vastaavat, eli esimerkiksi "useradd user1 -g nagcmd -p salasana". Ryhmän, esimerkissä määritelmä ”-g”, tulee olla sama kun nagios-käyttäjille asennuksessa luotu ryhmä, jotta luodulla käyttäjällä on luku- ja kirjoitusoikeus tarvittavaan dataan. Käyttäjien, esimerkissä "user1", salasana muutetaan komennolla "passwd user1".

5.2.2 Alustava konfigurointi

Ensimmäisiä toimenpiteitä ohjelmiston ja tämän vaatimien komponenttien käyttöönoton jälkeen on hallittavien laitteiden ja palveluiden seurannan konfigurointi. Nagios Coressa hallittavien laitteiden lisäys tehdään cfg-konfigurointitiedostoja editoimalla. Kyseiset tiedostot sijaitsevat yhteisessä kansiossa, jossa oletuksena tiedostot eri laitetyypeille kuten kytkimille, tulostimille ja Windows-pohjaisille käyttöasemille ja palvelimille. Kyseisiä laitemääritelmiä voi halutessaan luoda useita jokaista konfigurointitiedostoa kohden, mutta tässä tulee ottaa huomioon tiedoston rakenteen selkeys ymmärrettävyyden ja luettavuuden suhteen. Laajan tiedoston selaaminen ja tästä

tietyn laitteen löytäminen saattaa käydä vaivalloiseksi. Myös yksittäisen merkkivirheen olemassaolo aiheuttaa virheilmoituksen hallintaohjelmistoa ajettaessa, jolloin kyseinen virhe tulee löytää ja korjata.

Tiedostoja saa nimetä ja lisätä laitekohtaisesti, jolloin joka laitteelle voi halutessaan luoda oma konfigurointitiedosto. Tällöin tulee Nagios Coren nagios.cfg-tiedostossa määrätä kyseinen kansio kokonaisuudessaan käsiteltäväksi, eli jokainen cfg-päätteen omaava tiedosto määrätyssä kansiossa käsitellään. Jokainen laitetyyppi kannattaa tällöin asettaa omaan, nimeltään kuvaavaan kansioon.

#nagios.cfg

cfg_dir=/usr/loacl/nagios/etc/objects/switches
cfg_dir=/usr/loacl/nagios/etc/objects/servers
cfg_dir=/usr/loacl/nagios/etc/objects/routers

#j.n.e.

Muita vaihtoehtoja kansiorakenteelle on esimerkiksi sijainnin, osaston tai työntekijöiden vastuualueiden mukaan nimeäminen, mutta laitetyypin mukainen rakenne on kyseiselle verkonhallintajärjestelmälle optimaalinen ratkaisu. Aina kun Nagios Core käynnistyy tämä lukee konfigurointitiedostot yksittäiseen tiedostoon, jonka sijainti määrätään nagios.cfg-tiedostossa, esimerkiksi seuraavasti:

#nagios.cfg

object_cache_file=/usr/local/nagios/var/objects.cache

Laitteiden konfigurointitiedostot on mahdollista lukea yksittäiseen tiedostoon valmiiksi, josta Nagios Core käynnistyessä ajaa datan käyttöönsä, täten nopeuttaen käynnistysprosessia. Näin varsinaiset konfigurointitiedostot voivat olla käsittelyn alla vaikka nagios prosessia ajetaan alas ja takaisin käytiin. Nagios Corea käynnistettäessä komento-optiolla "-p" Nagios Core esiprosessoi laitteiden cfg-tiedostot muistiin, jolloin sammutettuna ohjelmiston käynnistys optiolla "-u" luetaan tiedot muistista, cfg-tiedostojen sijaan. Kyseisen objects.precache-tiedoston sijainti määrätään nagios.cfg-tiedostossa seuraavasti:

#nagios.cfg

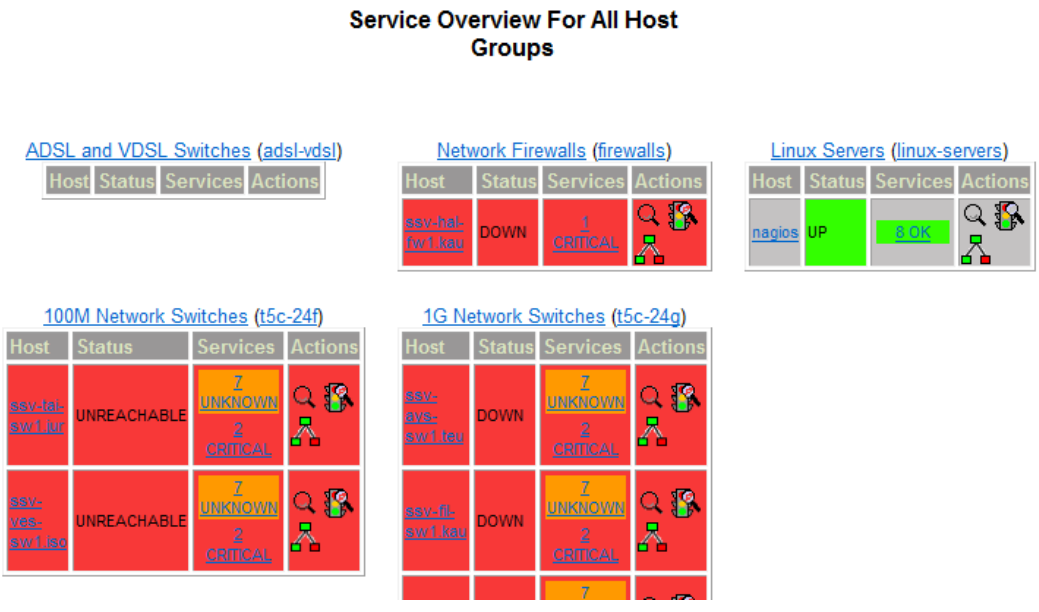
precached_object_file=/usr/local/nagios/var/objects.precache

Asennuksen yhteydessä luoduissa tiedostoissa on esimerkit laitteiden konfigurointiin. Seuraavana esimerkki yksittäisen kytkimen määrittelemisestä:

#switches.cfg

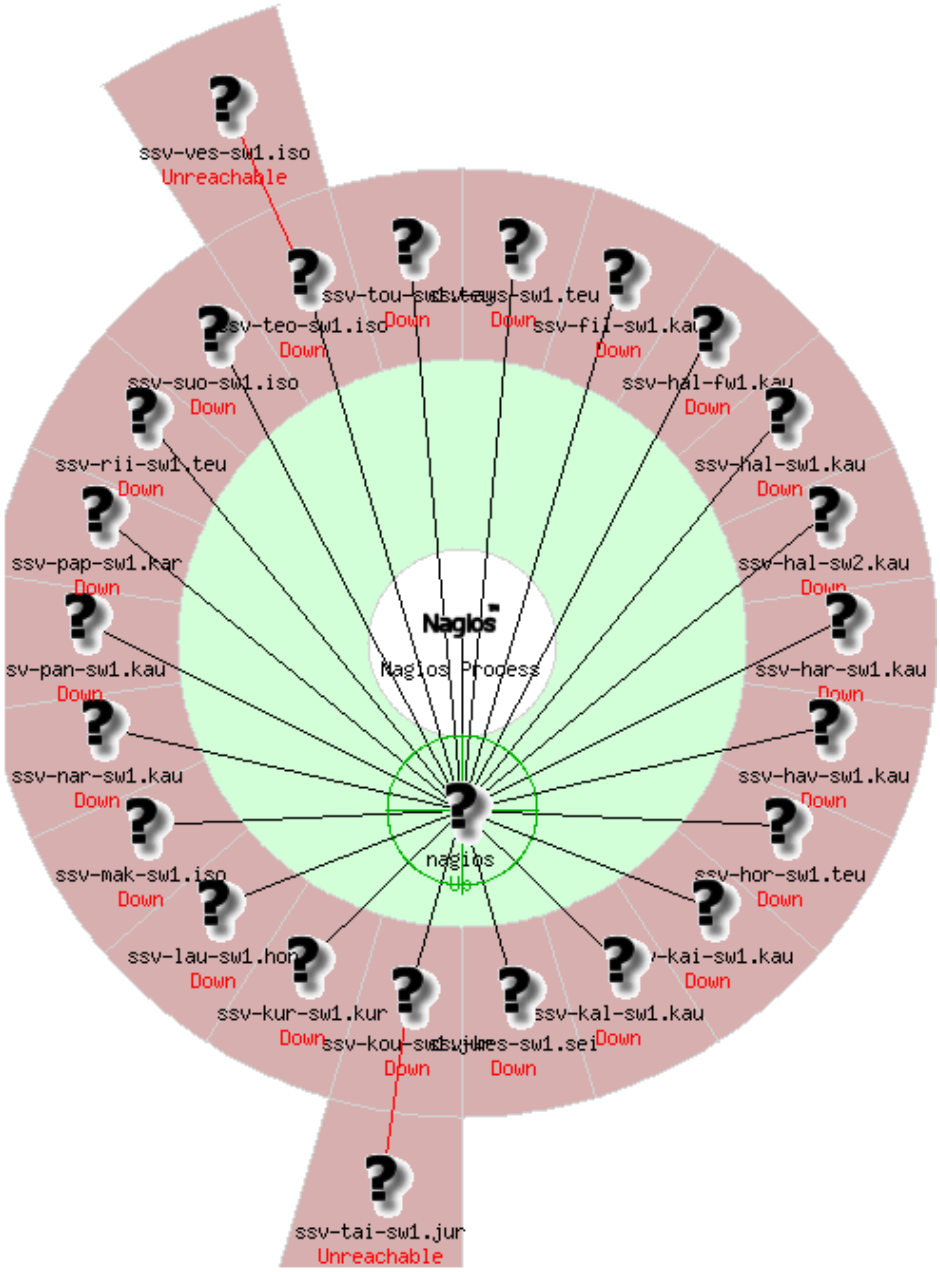
```
define host
{
use             generic-switch ;templatesta luettava data
host_name       switch-1                ;kutsumanimi
alias           Telco T5C-24G           ;laitteen nimi/malli
address         192.168.0.1             ;IP-osoite
hostgroups      t5c-24g                 ;laiteryhmä
parents         switch-0                ;vanhemmat
}
```

Yllä mainittuun laiteryhmään (hostgroup) kuuluu laitteita, joille määrätään käyttöön yhtenäiset valvontaparametrit. Eli ryhmään liitetään laitteita, joilla samoja valvottavia palveluita, ja näin näiden palveluiden tarkistuksessa voidaan kutsua kyseistä laiteryhmää. Laiteryhmien luonti luo myös ryhmittelyn käyttöliittymälle (kuva 4.)



Kuva 4. Laiteryhmät Nagios Coren käyttöliittymässä.

Samoin laitteiden palveluiden ryhmittely luo kuvainnollisen näkymän eri palvelutyypeistä. Laitteen hierarkkiset vanhemmat (parents) ovat ne laitteet, jotka luovat yhteyden Nagios Coren ja valvottavan laitteen välille. Jos joillekin lapsilaitteen (child) vanhemmista on havaittuna yhteysvirhe, kyseisen lapsen yhteyshäviöt eivät aiheuta hälytyksiä ja näkyy Nagios Coren Map-sivulla tilassa "Unreachable" (kuva 5.)



Kuva 5. Nagios Core Map.

Aiemmin mainitussa switches.cfg-tiedostossa, define host -määritelmässä, "generic-switch" on käytettävä malli, eli template, jota laitteelle käytetään. Template on yleisesti määriteltävissä laitteelle kyseisellä tavalla ja näin samoilla valvontakriteereillä oleville hallittaville laitteille saadaan nopeasti luotua yhtenäinen konfiguraatio. Kyseisessä templatessa ovat määriteltynä seuraavat:

```
#templates.cfg

define host
{
name                    generic-switch  ;tamplaten kutsumanimi
use                     generic-host            ;tälle käytettävä template
check_period   24x7             ;tarkistusajat
check_interval 5                ;tarkistusväli 5 min.
retry_interval  1               ;uudelleentarkistus 1 min.
max_check_attempts      10                      ;uudelleentarkistus lkm.
check_command           check-host-alive        ;komentokutsu
notification_period     24x7                    ;template ilmoitusajoille
notification_interval   120                     ;ilmoitusväli 2 h.
notification_options    d,u,r,f,s               ;ilmoitettavat tilat
contact_groups          admins                  ;ilmoitettavat ryhmät
register                0                       ;ei rekisteröidä templatea
parents                 nagios                  ;vanhemmat
}

define host
{
name                            generic-host
notifications_enabled           1               ;lähetetään ilmoituksia
event_handler_enabled           1               ;tapahtumankäsittelijät
flap_detection_enabled          1               ;tilanmuutoksen seuranta
notification_period             24x7
register                        0
}
```

Kyseisiä tietoja käytetään laitteen omien konfiguraatioiden lisäksi, laitteen hallinnassa, jos template näissä on määrätty. Näin useamman samaa templatea käyttävän laitteen hallintaprofiili saadaan päivitettyä hyvin yksinkertaisesti. Template määritellään Nagios Coreen paramertillä "register 0". Tämä ilmoittaa määritelmien olevan template, eikä varsinainen valvottava laite tai palvelu. Kyseisessä templatessa on määrätty laitteen tarkistusajat, eli ajat jolloin laitetta tulee valvoa. Käytetty määritelmä on "check_period". Määritelmällä tarkistusajat luetaan esimerkissä käytettävästä pohjasta "24x7", kirjattuna timeperiods.cfg-tiedostoon:

```
#timeperiods.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
}
```

Jos konfigurointitiedostossa ja templatessa ovat samat määritelmät, laitteen tiedot otetaan käyttöön tämän omasta tiedostosta ja templatessa olevat määritelmät hylätään. Templatea ei ole missään tapauksessa pakollista käyttää, vaan komennot voidaan määrätä myös pelkästään laitteen konfiguroinnissa.

Aiemmin kuvattua pohjaa käytetään myös ilmoitusaikojen määräämiselle. Pohjassa templates.cfg määrätty udelleentarkistusväli, esimerkissä tarkistus yhden minuutin välein, tulee käyttöön tilanteessa, jolloin laitteeseen ei saada yhteyttä, eli tämän tila on DOWN. Uudelleentarkistusten määrä määrää, montako kertaa tarkistus suoritetaan ennen kuin tapahtumasta luodaan ilmoitus (DOWN, HARD state). Tällöin huomioidaan tilat, joista ilmoitus tulee lähettää. Esimerkiksi "d, u, r, f, s" (taulukko 1.)

Taulukko 1. Ilmoitustilatyypit.

d = DOWN	**Laite alhaalla**
u = UNREACHABLE	**Yhteys laitteeseen poikki**
r = RECOVERIES (OK state)	**Palautuu tilasta joka muu kuin OK**
f = FLAPPING	**Jatkuva tilan muutos**
s = scheduled downtime (start/end)	**Laitteen huolto alkaa tai loppuu**
n = none	**Ei lähetetä ilmoituksia**

Ilmoitusta lähetettäessä ilmoitettaville yhteyshenkilöille, nämä yhteystiedot haetaan käytössä olevasta templatesta yhteyshenkilöryhmistä, määriteltynä tiedostossa contacts.cfg. Määrättynä ovat yksittäiset kontaktit ja ryhmät joihin nämä kuuluvat.

```
#contacts.cfg

define contact
{
contact_name            ssv                     ;kutsumanimi
```

```
use                     generic-contact         ;tälle käytettävä template
alias                   Kenth Joki              ;koko nimi
email                   kjoki@ssv.fi            ;sähköpostiosoite
}
define contactgroup
{
contactgroup_name       admins                  ;kutsumanimi
alias                   Nagios Administrators   ;ryhmän nimi
members                 ssv                     ;ryhmän kontaktit
}
```

Käytetyssä tamplatessa "generic-host", tiedostossa templates.cfg, on määrättynä esimerkiksi seuraavat:

```
#templates.cfg

define contact
{
    name                        generic-contact
    service_notification_period 24x7
    host_notification_period    24x7
    service_notification_options w,u,c,r,f,s
    host_notification_options    d,u,r,f,s
    service_notification_commands notify-service-by-email
    host_notification_commands   notify-host-by-email
    register                    0
    }
```

Templatessa on määrättynä ilmoitusajat laitteiden (host) ja näillä olevien valvottavien palveluiden (service) tiloille (kuva 6), kuten myös kyseiset tilat joista ilmoittaa (taulukko 2.)

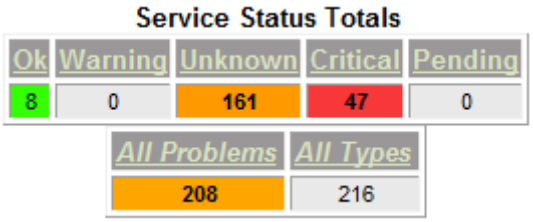
Taulukko 2. Hälytystilatyypit.

w = WARNING	**Varoitus palvelun tilasta**
u = UNKNOWN	**Tarkistusta ei voida suorittaa**
c = CRITICAL	**Kriittinen tilan muutos**
r = RECOVERIES (OK state)	**Palautuu tilasta joka muu kuin OK**
f = FLAPPING	**Jatkuvasti vaihteleva tila**
n = none	**Ei lähetetä ilmoituksia**

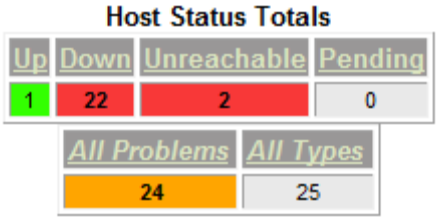
5.2.3 Tilatyypit ja tapahtumankäsittelijät

Laitteiden ja palveluiden SOFT- ja HARD-tilat määräytyvät ja näkyvät tarkastusten hetkellisten ja varmennettujen tilojen mukaan (kuvat 6 ja 7.) Soft error -states ovat muut kuin OK- tai UP-tilat, joiden havainto on tehty mutta tilan tarkastuksia ei ole vielä suoritettu kirjattua määrää (max_check_attempts).

Siirtymä CRITICAL SOFT -tilaan tapahtuu, kun virhe on havaittu ja WARNING SOFT -tilaan, kun esimerkiksi kaksi tarkistusta on suoritettu samalla tuloksella. CRITICAL HARD -tila (hard error-state) on tila, johon laite tai palvelu luokitellaan kun kaikki tarkistukset ovat suoritettuina. WARNING HARD -tilaan siirtymä tapahtuu, kun mahdolliset kirjatut tapahtumankäsittelijät on suoritettu, ilmoitukset on lähetetty ja tila pysyy samana seuraavalla tarkistuksella.

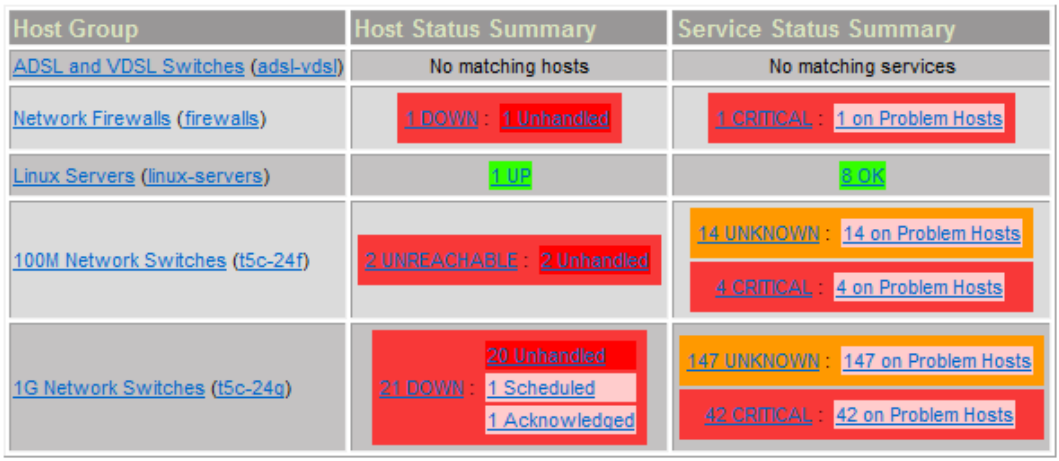


Kuva 6. Palveluiden tilojen kooste.



Kuva 7. Laitteiden tilojen kooste.

Tilan korjaannuttua palvelu tai laite siirtyy OK HARD -tilaan (hard recovery-state), tälle määrätyt mahdolliset tapahtumankäsittelijät toteutuvat ja recovery-ilmoitus lähetetään. Jos tila korjaantuu OK- tai UP-tilaan ennen tarkistusten tekoa, palvelu tai laite siirtyy soft recovery-tilaan ja tarkistusten lukumäärä resetoituu. Tämän jälkeen palvelu tai laite siirtyy välittömästi OK HARD -tilaan. Palautuminen (soft recovery-state) ei aiheuta recovery-ilmoituksia, koska kuvattussa tapahtumassa ei havaittu varmennettua häiriötä. Kuten mainittu, ilmoitusten lähetys määräytyy laitteiden ja palveluiden cfg-tiedostojen mukaan ja ilmoitusten vastaanottaminen määräytyy kontaktien cfg-tiedostojen mukaan. Kuvatut tilojen kooste näkyy myös laiteryhmittäin käyttöliittymällä (kuva 8.) /19/



Kuva 8. Laitteiden tilojen kooste ryhmittäin.

Tilansiirtymien aiheuttamat tapahtumankäsittelijöiden ajot (event-handlers) kuten ilmoitusten lähetys (notifications) on määrättävissä laite- ja palvelukohtaisesti, ilmoituksille ”notification-options”, kun nagios.cfg-tiedostossa nämä ovat määrätty käyttöön, eli "eventhandlers=1". Nagios Corella on kahdenlaisia tapahtumankäsittelijöitä. Globaalit ja laite- tai palvelukohtaiset host- ja service-tapahtumankäsittelijät. Globaalit käsittelijät ajetaan näitä kutsuttaessa, aina ennen laite- tai palvelukohtaisia, jotka suoritetaan heti tämän jälkeen.

Event-handler on määrätty ohjelmallinen toteutuma, jonka Nagios Core käynnistää, tilanmuutoksen sitä kutsuessa. Nämä suoritetaan esimerkiksi kun laitteen tai palvelun tila on soft-ongelmatilassa, siirtyy hard-ongelmatilaan tai palautuu kyseisistä tiloista. Esimerkkeinä event-handlereistä voidaan mainita rikkoontuneen palvelun uudelleenkäynnistys, virheilmoituksen lähetys tai tapahtuman kirjaus lokitiedostoon.

Globaalit host- ja service-tapahtumankäsittelijät määrätään nagios.cfg-tiedostossa ja laite- ja palvelukohtaiset laitteen ja palvelun määritelmässä.

#nagios.cfg

global_host_event_handler=<command>
global_service_event_handler=<command>

#switch.cfg

event_handler command_name

5.2.4 Tarkistukset

Laitteille ja näiden palveluille määrätään kutsut tarkistusprosesseille laitteen omassa konfigurointitiedostoissa tai erillisessä konfigurointitiedostossa commands.cfg. Kyseinen tiedosto on yhtenäinen template-tiedosto tarkistuskomentokutsuille, josta tarkistus tarpeen mukaan voidaan ottaa käyttöön, kutsumalla tätä template-komentoa laitteen cfg-tiedostossa.

#command.cfg

#check-host-alive command definition

define command
{
command_name check-host-alive
command_line $USER1$/check_ping -H $HOSTADDRESS$ -w 3000.0,80% -c 5000.0,100% -p 5
}

Yllä olevassa esimerkissä tarkistetaan verkonhallintapalvelimen yhteyttä laitteeseen ping-komennolla. Kyseistä komentoa tulee kutsua jokaisella laitteella, jotta näiden yhteys saadaan todettua ja näiden valvontaa ylläpidetään.

Tarkistuskomennot suoritetaan makroilla, tiedostoviitteillä ja raja-arvojen määritelmillä. Tiedostoviite osoittaa tiedostoon, joka on nimetty tarkistuksen mukaan, eli tässä tapauksessa "check_ping". Näiden oletuspolku on /usr/local/nagios/libexec/, kuten kirjattuna nagios.cfg-tiedostossa. Tarkistus-komennon ajossa makro korvataan tämän viittaamalla todellisella arvolla tai merkkijonolla. Makrojen merkitys määrätään tiedostossa "resource.cfg".

#resource.cfg

$USER1$=/usr/local/nagios/libexec

Mainittu tarkistus "check-host-alive" osoittaa tiedostopolun **”**/usr/local/nagios/libexec” tiedostoon check_ping. Tiedostossa kirjatun tarkistuksen vaatimat kriteerit ovat laitteen IP-osoite (-H=hostname), rajaparin

kriteerit WARNING- ja CRITICAL-tiloille (-w ja -c) ja lähetettävät paketit (-p). Rajaparit ovat maksimi RTA millisekunneissa (3000ms/5000ms) ja prosentuaalinen pakettihäviö (80%/100%). Vastaavia tarkistusajoja on Nagios Coressa kattava määrä ja yhteisön luomat tarkistukset täydentävät muutamaa puutetta, kuten porttien kuormituksen seurantaa.

Makrojen käyttö on myös suositeltavaa aran tiedon kuten salasanojen ja SNMP-ryhmien käytössä. Tietoa haettaessa resource.cfg-tiedostosta tämä ei tule nähtäväksi Nagios Core CGI-käyttöliittymälle, toisin kuin kuvassa 9 saattaisi nähdä. Tässä SNMP-ryhmä on kirjattuna palvelun tarkistusta kutsuessa.

Service

Host	Description	Max. Check Attempts	Normal Check Interval	Retry Check Interal	Check Command	Check Period
ssv-pap-sw1.kar	Uptime	3	0h 10m 0s	0h 2m 0s	check_snmp!-P 2c -C [redacted] -o system.sysUpTime.0	24x7

Kuva 9. Palvelun tarkistus Nagios Core CGI:llä.

Laitteiden palvelut ja näille suoritettavat palvelutarkistukset määrätään laitteen konfigurointitiedostossa, esimerkiksi switch.cfg-tiedostossa, seuraavasti:

```
define service
{
use                     generic-service
hostgroups              t5c-24g,t5c-24f
service_description     Port 1 Link Status
check_command           check_snmp!-P 2c -C $USER2$ -o
                        ifOperStatus.1$
```

Kyseisessä esimerkissä palvelutarkistus, joka on määrättynä laitteille, jotka ovat ryhmien "t5c-24g" ja "t5c-24f" alaisia, suorittaa SNMP-tarkistuksen kytkimen hallinta-agentille portin numero 1 yhteydestä. Pluginin määritelmien mukaisesti agentin lähetettäessä arvon "1", palvelu määrätään OK-tilaan. Jos agentin lähettämä arvo on muuta kuin "1", palvelun tilaksi tulee "CRITICAL". Mainittu palvelutarkistus käyttää templatea "generic-service", jossa on kirjattuna yleisimmät halutut määritelmät seuraavasti:

#templates.cfg

}
define service{

name	**generic-service**	
active_checks_enabled	**1**	**; aktiivinen tarkistus**
passive_checks_enabled	**1**	**; passivinen tarkistus**
parallelize_check	**1**	**; vierekkäiset tarkistukset**
obsess_over_service	**1**	**; pakkomielle palvelusta**
notifications_enabled	**1**	**; ilmoitukset palvelusta**
event_handler_enabled	**1**	**; tapahtumankäsittelijät**
flap_detection_enabled	**1**	**; tilanmuutosten seuranta**
failure_prediction_enabled	**1**	**; virheiden ennakointi**
retain_status_information	**1**	**;tallennetaan tilatieto**
retain_nonstatus_information	**1**	**;tallennetaan palveludata**
is_volatile	**0**	**;volatiili palvelu**
check_period	**24x7**	
max_check_attempts	**3**	
normal_check_interval	**10**	
retry_check_interval	**2**	
contact_groups	**admins**	
notification_options	**w,u,c,r**	
notification_interval	**60**	
notification_period	**24x7**	
register	**0**	

}

Aktiivinen tarkistus määrää palvelun vakituisen ajan välein suoritettavaksi. Nämä ovat Nagis Coren suorittamia ja käsittelemiä. Passiiviset tarkistukset ovat ulkoisen ohjelmiston suorittamia, jolloin tämä kirjaa datan tiedostoon, josta Nagios Core tämän käsittelee, jos näin on määrätty. Vierekkäiset tarkistukset mahdollistaa, Nagios Coren jatkuvan valvontatoiminnan tukemiseksi, useamman toiminnon samanaikaisen suorituksen. Jos kyseinen komento poistetaan käytöstä, Nagios Core ei suorita muita toimenpiteitä ennen kuin kyseinen tarkistus on ajettu loppuun. Useamman samanaikaisesti suoritettavan prosessin ajo rasittaa verkonhallintapalvelinta, mutta tällä on voimakas vaikutus Nagioksen toiminnallisuuteen ja tulee olla käytössä kaikilla suoritettavilla palvelutarkistuksilla.

"obsess_over_service" -komento määrää laitteen tai palvelun, niin sanotun, pakkomielteisen valvonnan alaiseksi. Tällöin tälle määrätty ocsp-komento ajetaan aina laite- ja palvelutarkistuksen jälkeen.

#commands.cfg
#obsessive compulsive service processor command

ocsp_command=<command>

Jos kyseinen pakkomielteinen valvonta on käytössä laitteesta tai palvelusta lähetetään hälytykset heti kaikkien oleellisten tilamuutosten jälkeen, vaikka näin ei olisi muuten määrätty. Vastaava määritelmä on kirjattavissa laitekohtaisesti (ochp, obsessive compulsive host processor command). Kyseiset määritelmä eivät ole oleellisia kyseisen verkonhallintapalvelimen ja tämän nykyisen valvottavan verkon kokoonpanon seurannassa, mutta verkon mahdollisen laajennuksen aiheuttama kuormitus hallintapalvelimelle voi jatkossa vaatia tämän ochp-määritelmän käyttöönottoa. Tällöin jokainen hajautetulla Nagios Core-palvelimella suoritetun tarkistuksen ajon tulos voidaan ocsp-komennolla lähettää keskeisen verkonhallintapalvelimen käsiteltäväksi, jakaen kuormitusta useamman palvelimen välillä (distributed monitoring). Tämä on toteutettavissa Nagios Coreen integroidulla nsca-lisäkomponentilla. /20/

Palvelinohjelmiston käynnistyessä tämän suorittamat palvelutarkistukset saattavat olla suuressa verkossa huomattavat, jolloin myös tämä aiheuttaa rasitusta verkonhallintapalvelimelle. Tarkistusten ajallinen hajonta auttaa suorituksessa, jolloin samanaikaisten tarkistusten määrä vähenee. Kyseinen hajonta on määrättävissä seuraavasti:

#nagios.cfg

#n=none
#d=dumb 1 sec delay
#s=smart inter-check delay calculation
#x.xx=inter-check delay of x.xx sec

service_inter_check_delay_method=s ;hajontamalli
max_service_check_spread=30 ;maksimi hajonta-aika käynnistyksestä

Resurssienhallinnan kannalta oleellista on myös SNMP-kyselyiden optimointi esimerkiksi kyselyn yhteydessä määräämällä osoittaja tarvittavalle MIB-tietokannan datalle. Näin ei ole tarvetta ladata koko tietokantaa, joka palvelu-tarkistuksen yhteydessä.

check_command check_snmp!-C public -o ifOperStatus.1 -r 1 -m RFC1213-MIB

Nagios Coren esittämän laitteiden ja palveluiden tiladatan lisäksi tämä mahdollistaa erillisten pluginien käsittelyn suoritusdatasta (performancedata), jolloin nämä palauttavat tiedon käyttöliittymälle esiteltäväksi. Nagios Core määrätään, nagios.cfg-tiedostossa, kirjaamaan suoritusdatan erilliseen tiedostoon, esimerkiksi tiedostopolulla "/var/nagios/perfdata.log". Kyseisestä lokitiedostosta erillinen ohjelma käsittelee tiedon, ja jos näin määritelty, palauttaa tämän käyttöliittymälle esilteltaväksi, esimerkiksi kuvaajan muodossa.

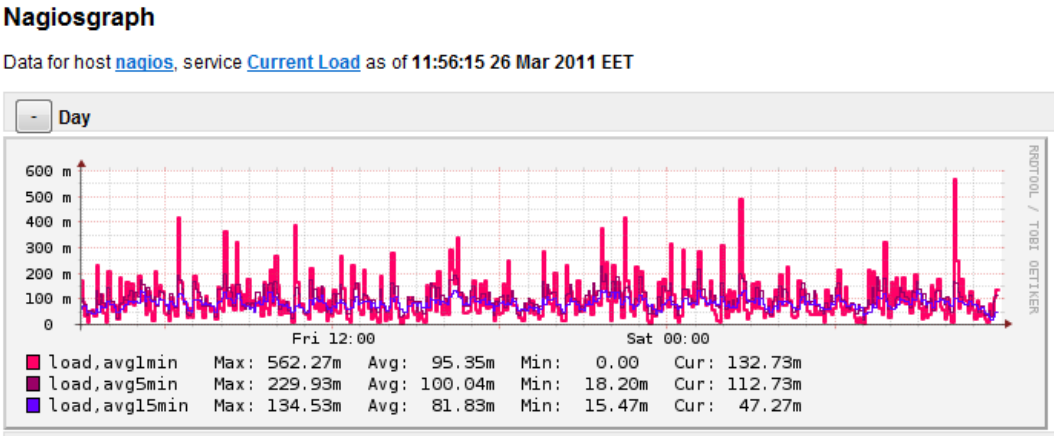
```
#nagios.cfg

process_performance_data=1
service_perfdata_file=/var/nagios/perfdata.log
service_perfdata_file_template=$LASTSERVICECHECK$||$HOSTNAME$||$SERVICEDE
SC$||$SERVICEOUTPUT$||$SERVICEPERFDATA$ service_perfdata_file_mode=a
service_perfdata_file_processing_interval=30
service_perfdata_file_processing_command=process-service-perfdata
```

Kyseiset määritelmät komentavat Nagios-prosessin kirjaamaan pluginin keräämän vastaanotetun tiedon tiedostoon "perfdata.log", määritellyn templaten mukaisesti. Esitelty template on kuvaajia luovan nagiosgraph-sovelluksen tarpeiden mukainen. perfdata.log-tiedostoon lisätään (a=append), 30. sekunnin välein, kerätty tieto aiemmasta seuraavalle riville. Palvelulle, tämän cfg-tiedostossa määrättävä, komento on "process-service-perfdata=1". Suupohjan Seutuverkko Oy:lle toteutettavassa verkonhallintapalvelimessa oleellinen vastaava data on esimerkiksi kytkinten, suorittimen ja porttien kuormitus, kuten myös kytkinten saavutetavuus Nagios-tarkistuksille. Saavutettavuus tarkistetaan PING-ohjelmistolla.

5.2.5 Kuvaajat

Oleellinen osa toimivaa hallinnointia on pitemmän ajan verrannollinen laadun seuranta. Varsinkin, asiakaskannan kasvun seurauksesta, alati laajenevassa tietoliikenneverkossa tapahtuva kuormituksen lisäys palvelimelle tulee olla selkeästi nähtävissä. Tälle ja muille valvottaville palveluille tulee myös olla määrättynä hyväksyttävät raja-arvot, asiakkaalle myytävän palvelun laadun varmistamiseksi. Kyseinen on Suupohjan Seutuverkko Oy:lle suunnitellussa toteutuksessa saavutettu, Nagios Coren yhteisön luomalla, Nagiosgraph-

sovelluksella. Kyseinen sovellus integroidaan Nagios Core -prosessiin ja tämän käyttöliittymään, varsinaisen Nagiosgraph-sovelluksen asennuksella ja Nagioksen konfigurointitiedostojen määräyksiä editoimalla. Kyseinen sovellus vaatii Nagios Core -toteutuksen kirjoittavan tiedot seuratuista palveluista, sivulla 34 mainittuun perfdata.log-tiedostoon. Nagiosgraphin sisältämät suoritusdatan prosessointikomennot hakevat kirjatun tiedon tiedostosta verrattavaksi sen map-tiedostojen parametreihin. Kyseisistä map-tiedostoista sovellus hakee perfdata.log-tiedostosta haetun tiedon mukaisen viitetiedon ja Nagiosgraph kirjaa tiedon viitteen mukaisesti tietokantaan, sopivan kyseisen referenssitiedon löydettyä. Nagiosgraph-asennus sisältää Nagios Coren CGI-skriptien kansioon sisällytettävän show.cgi-tiedoston, jota kutsuessa, Nagioksen käyttöliittymällä on nähtävissä tietokannan datan mukaisesti luotu kuvaaja kyseiseltä palvelulta kerätystä tiedosta (kuva 10.)



Kuva 10. Kuvaaja käyttöliittymällä.

Kyseisen kuvaajan haetaan käyttöliittymältä palvelukohtaisesti cfg-tiedostoissa määrätyillä komennoilla kuten esimerkiksi:

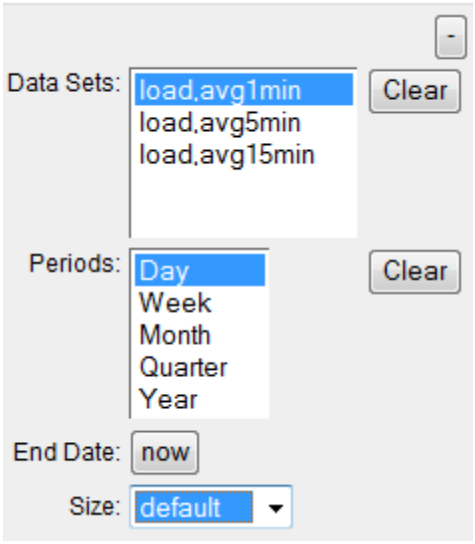
#localhost.cfg

#action_url -komento

/nagios/cgi-bin/show.cgi?host=$HOSTNAME$&service=$SERVICEDESC$

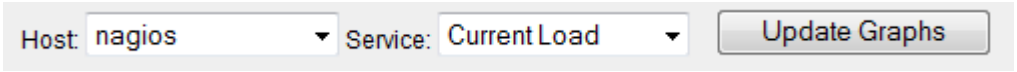
Kyseinen komento näkyy CGI-skriptien kautta käyttäjän selaimessa kuvakkeella merkittynä "Perform Extra Action" -linkkinä. Kuvaajan esitysvaihtoehdot ovat

määrättävissä näytettävän ajanjakson, kuvaajan koon ja palvelulta kerätyn datan mukaan (kuva 11.)



Kuva 11. Kuvaajan esitysmääritelmät.

Käyttöliittymällä esiintyy myös valikko seurattavien laitteiden ja näiden hallittavien palveluiden kuvaajien hakuun (kuva 12.)



Kuva 12. Kuvaajan valinta.

Nagiosgraph on riippuvainen useasta lisäkomponentista. Tietokanta, jota sovelluksen luoma yhteisön jäsen on päättänyt hyödyntää, on RRDtool-sovelluksen Round Robin Database tarjoama tietokanta. Tämä on oikeaoppisesti konfiguroituna hyvä, palvelimelle vähäisen rasituksen aiheuttama sovelluspaketti. Järkevien, vanhemman tiedon, arkistointimääritelmien ansiosta saadaan palvelutarkistuksista kerätty tieto pakattua tilavaatimuksiltaan pieneen tiedostoon. Nagiosgraph-sovellus luo rrd-tiedoston jokaista tarkistettua palvelua kohden. Tiedoston saavutettua määrätyn maksimikoon, tieto arkistoidaan ja tiedostoa kierrätetään kirjoittamalla jo arkistoidun datan yli. Muihin Nagiosgraphin riippuvuuksiin voidaan luetella tämän toteutuksen ohjelmointikielen Perlin tuki

kuten myös jo, Nagios Coren vaatimusten mukaisesti, asennetun CGI-paketin Perl-komponentit. /16/

Suupohjan Seutuverkko Oy:lle tuotteistetulla verkonhallintapalvelimella oleellisimmat kuvaajat ovat palvelimen kuormitus, prosessien määrä ja kovalevytila, eli Load, Total Processes ja Root Partition, kuten myös kytkinten välisten yhteyksien laatu ja liittymien tiedonsiirtonopeus. Porttien tiedonsiirto saadaan suorittamalla SNMP-kysely kytkimen agentille kyseisen liittymän lähettämien ja tälle tulevien oktettien määrästä. Kyselyssä hyödynnetään pluginia check_snmp_int.pl, joka on ohjelmallisesti määrätty suorittamaan laskennalliset toimenpiteet, suoritetulla kyselyllä saadusta datasta. Kyselyt antavat ainoastaan tiedot lähtevästä ja vastaanotetusta määrästä oktetteja, mutta kahden kyselyn aikajako ja kyselyajankohdittain lähetetyt ja vastaanotetut oktetit antaa, laskuoperaation suorittamalla, portin tiedonsiirtonopeuden (download/upload). Plugin täytyy kopioida libexec-kansioon, ja tiedostoon commands.cfg täytyy määritellä pluginin mukainen kyselyrakenne. Kyselyrakenne määrätään hyödyntämään resource.cfg-tiedostoon määrättyjä makroja, kappaleessa ”Tarkistukset”, sivulla 31 esitetyllä tavalla.

#commands.cfg

```
define command
{
command_name            check_snmp_int_v1
command_line            $USER1$/check_snmp_int.pl -H
                        $HOSTADDRESS$ $USER2$ -n $ARG1$ $ARG2$
}
```

Palvelutarkastuksessa käytetään pluginin määritelmien mukaisia komentoja, esimerkiksi seuraavasti:

#switch.cfg

```
define service
{
use                     generic-service
hostgroups              t5c-24g
service_description     check_int_eth0_bdw
check_command           check_snmp_int_v3!eth0!-k -w 100,50 -c 0,0
```

process_perf_data 1
}

Suoritettavan kyselyn antaman datan maksimi-raja-arvot liitännälle "eth0" määrätään vastaanottonopeudelle sataan kilotavuun ja lähetysnopeudelle viiteenkymmeneen kilotavuun. Kriittisiä (CRITICAL) raja-arvoja ei määrätä. Jotta kyselystä saadaan kuvaajat Nagiosgraph-ohjelmistolla, täytyy tälle olla määrättynä dataan verrannollinen map-tiedosto.

5.2.6 Sähköposti-ilmoitukset

Ilmoitusten lähetys oleellisista tapahtumista on myös tärkeä ominaisuus käytännöllisessä verkonhallintapalvelimessa. Eli palvelimeen tulee integroida SMTP-palvelimen ominaisuudet, kuten myös itse sähköpostin luontiin vaadittu ohjelmisto. Suupohjan Seutuverkolle toteutetussa verkonhallintapalvelimessa kyseinen on saavutettu vapaan lähdekoodin Postfix, Simple Mail Transport Protocol-palvelinohjelmistopaketilla. Kyseinen sisältää sähköpostin lähetysohjelmiston ja välityspalvelimen ominaisuudet, joilla saadaan ilmoitukseen tapahtumakohtaiset tiedot, esimerkiksi seuraavanlaisen sähköpostin muodossa:

******* Nagios *******

Notification Type: PROBLEM
Host: ssv-kal-sw1.kau
State: DOWN
Address: x.x.x.x
Info: CRITICAL - Host Unreachable (x.x.x.x)

Date/Time: Mon Mar 28 11:31:07 EEST 2011

Postfix MTA (Mail Transport Agent) -ohjelmisto mahdollistaa, esimerkin mukaisen, sendmail-komennolla luodun sähköpostin välityksen kaikista Nagios Coreen määritellyistä ilmoitettavista tapahtumista. Ilmoituksen luonti suoritetaan Nagios Coren tapahtumankäsittelijän luodessa ilmoituksen rakenteen ja sisällön, määrätyn komennon mukaisesti. Komento sisältää otsikon, sisällön, sähköpostiohjelmiston ja vastaanottajien määritelmät. Makroja käyttämällä saadaan hyödynnettyä samaa rakennetta kaikille laitteille ja palveluille, esimerkiksi seuraavasti:

```
#commands.cfg
# 'notify-host-by-email' command definition

define command
{
command_name    notify-host-by-email

command_line    /usr/bin/printf "%b" "Subject:**
$NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$
**\n***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState:
$HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo:
$HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" |
/usr/sbin/sendmail $CONTACTEMAIL$
```

MTA-ohjelmistolle tulee määrätä hyväksyttävät yhteydet, jotka kyseisen toteutuksen mukaisesti ovat verkon sisäiset osoitteet, pääosin itse verkonhallintapalvelin, jonka IP-osoite on 127.0.0.1.

6 TULOKSET

Toteutunut verkonhallintapalvelin sijaitsee nykyiseltään Kauhajoella, Suupohjan Seutuverkko Oy:n toimitiloissa, valokuituverkkoon liittämättömänä. Palvelimen ollessa yhteydessä kyseiseen verkkoon, tämän toiminta oli hyväksyttävää, vaikkakin osittain puutteellista. Tavoitteet, Suupohjan Seutuverkko Oy:lle tuotteistetulle verkonhallintapalvelimelle, olivat käyttäjäystävällisyys, käyttäjäympäristön graafisuus tärkeimpänä, yhteyksien tilojen seuranta, yhteyksien SLA-arvon laskeminen ja esittäminen, verkon ja tämän yhteyksien kuormitustiedon takautuva arkistointi, hälytysten automaattinen lähetys sähköpostin ja tekstiviestin välityksellä, kuten myös lokitiedostojen yhtenäinen keruu ja näiden yksinkertaisettu haku ja luku.

Tavoitteiden toteutumat jäivät monilta osin puutteellisiksi. Konfigurointiympäristön ollessa komentokehotepohjainen, seurattavan laitteiston ja näiden palveluiden konfigurointitiedostojen määritelmien suhteen, käyttäjäystävällisyys ei tältä osin toteutunut. Kytkinten ja näiden yhteyksien kuormituksen seuranta jäi kaipaamaan parannusta, etenkin tämän esitysmuodon suhteen. Nykyiseltään toteutuman luomat kuvaajat eivät esitä tarpeaksi tarkkaa tietoa, tämän tiedon ollessa ainoastaan suuntaa antavaa. Kuvaajien esitysmuodon tulisi myös olla mukautettavampi. Tässä jäätiin kaipaamaan määritellyn päivämäärävälin mukaisesti piirrettyjen kuvaajien esitysmahdollisuutta. Hälytysten lähetys tekstiviestillä, GSM-verkossa ja lokitiedostojen keskitetty keruu ei toteutunut, vaikkakin lokitiedostojen keruulle löydettiin teoreettisesti toimiva, ohjelmallinen ratkaisu (katso Syslog-ng, sivu 42). Tekstiviestien lähetys on myös toteutettavissa nykyiselle verkonhallintapalvelimelle (katso sivu 43).

Vaatimuksien pohjalta tehdyt, toimivasti toteutuneet ratkaisut ovat verkon toiminnan seurannan käyttäjäystävällisyys, yhteyksien tilojen ja näiden muutosten valvominen ja kyseisen tiedon arkistointi, SLA-arvon lasku ja esitys ja ilmoitusten ja hälytysten lähetys sähköpostilla.

7 JOHTOPÄÄTÖKSET

Itse ohjelmistossa ei ole varsinaisia puutteista, ainoastaan käyttämättömiä ominaisuuksia. Suupohjan Seutuverkko Oy:lle tuotteistetussa verkonhallintapalvelimessa esiintyvät toiminnan vajaavaisuudet ovat toteutuksesta johtuvia, ja tutkimuksella ja työllä korjattavissa. Jo aiemmin mainittu kattava yhteisö on jatkuvasti aktiivinen luomaan omia toiminnallisia ratkaisuja aina tarpeen nähdessä, ja jo olemassa olevia pluginejä ja rinnakkaisia sovelluksia käyttäen Nagios Core on täydellisesti tarkoituksenmukainen verkonhallintapalvelinratkaisu. Suupohjan Seutuverkko Oy:n hallittavan laitteiston ollessa pääosin erilaisia kytkimiä, laitteistolla on yhtenäisiä ominaisuuksia ja samankaltaiset valvonta- ja hallinnointitarpeet, ja näin hallintapalvelimen konfiguroinnin ja varsinaisen käyttöönoton jälkeen toteutus on suhteellisen vähällä vaivalla laajennettavissa kattamaan koko verkon, nykyiseltä ja tulevalta laajuudeltaan.

Oleellisin nykyisen toteutuksen puutteista on lokien keskitetty välitys laitteiston ylläpitäjän luettavaksi. Vaatimusten mukaisesti palvelinohjelmistosta ja kytkimistä kerättyjen lokitiedostojen sisältö tulisi saada keskitetysti, helposti luettavaan muotoon, mieluiten selainpohjaiselle käyttöliittymälle tai vaihtoehtoisesti sähköpostilla asianmukaisille henkilöille. Teoreettisesti tämä on nykyisellä ohjelmistolla toteutettavissa aiemmin mainituilla tapahtumankäsittelijöillä (eventhandlers), määräämällä tiettyjen tapahtumien sattuessa Nagios Coren lähettämään lokitiedoston sisällön sähköpostilla asianmukaisiin osoitteisiin, esimerkiksi hälytysilmoituksen yhteydessä tai määrätyin väliajoin. Lokitiedostojen läpikäynti on toki erittäin vaivalloista ilman lokitiedolle omistettua tietokantaa.

Lokitiedostojen keskitettyyn lukuun on myös jo luotuja, Nagioksen rinnalla toimivia, vapaan lähdekoodin lisenssin alaisia sovelluksia. Esimerkkinä näistä mainittakoon Syslog-ng. Tämä kerää lokitiedostojen sisällön yhtenäiseen tietokantaan, josta kyseinen tieto on luettavissa selainpohjaisella käyttöliittymällä.

Tietoa saa hakea ajanjakso- ja sisältökohtaisesti. Ohjelmistolle luvataan päteviä tietoturvaominaisuuksia ja käyttäjäystävällista käyttöliittymää. /17/

Myös hälytysten ilmoitus SMS-viestillä, GSM-verkossa, on toivottava ominaisuus Suupohjan Seutuverkko Oy:n verkonhallintapalvelinohjelmistolle. Nykyisten, varsinkin kyseisellä alalla toimivien työntekijöiden ja toimihenkilöiden, matkapuhelimien osatessa vastaanottaa sähköpostilaatikkoon tulevia sähköposteja, tämä on verrannollisesti toissijainen toiminnallisuus, mutta kuitenkin toteutettavissa myös kyseiselle palvelimelle. Tämä toki vaatii verkonhallintapalvelimelta yhteyden SMS-viestejä lähettävään laitteeseen kuten GSM-puhelimeen tai –modeemiin, sekä yhteensopivan SMS-gateway-ohjelmiston kyseiselle laitteelle. /18/

Yrityksen ottaessa verrannollisen valmiiksi tuotteistetun varkonhallintapalvelimen käyttöönsä, salausmenetelmien hyödyntäminen ja niiden mahdollinen vahvistaminen on myös varteenotettava käytäntö. Etenkin SSH-tunneloinnille ja SMTP-palvelinohjelmistolle olisi huomattavasti vahvemmat salausmenetelmät saatavilla.

Ohjelmiston mukautettavuus on tämän suurimpia etuja, jolloin sovellus ja käyttöliittymä saadaan toiminnoiltaan sekä ulkomuodoltaan vastaamaan yrityksen tarpeita ja ilmettä. Myös Nagios Core Map, kuvattuna sivulla 26, on mukautettavissa, joskaan tätä ei ulkomuodoltaan saa täysin vastaamaan Suupohjan Seutuverkko Oy:n kytkinrakennetta. Joitakin yritykselle tärkeitä valvontaominaisuuksia on käytössä heti Nagios Coren asennuksen jälkeen, ilman valvontaprosessien erillistä käyttöönottoa, esimerkiksi SLA-arvo, kuvattuna sivulla 9. Toteutusta en kuitenkaan suosittelisi kyseisen pienehkön yrityksen käyttöön, ajatellen ohjemiston vaativuutta henkilöstöresursseja kohtaan. Maksullisten, huomattavasti käyttäjäystävällisempien verkonhallintaohjelmistojen kustannukset säästyvät osittain jo työntekijöiden palkkauskustannuksista.

LÄHTEET

/1/ Open Source Initiative. The Open Source Definition [Viitattu 14.10.2010] Saatavilla Internetissä: <URL:http://www.opensource.org/docs/osd>

/2/ DPS Telecom. ONS 15454 Reference Manual R8.5.x – SNMP [Viitattu 25.4.2011] Saatavilla Internetissä: <URL:http://www.dpstele.com/layers/l2/snmp_l2_tut_part1.php>

/3/ Cisco Systems Inc. ONS 15454 Reference Manual R8.5.x -- SNMP [Viitattu 14.2.2011] Saatavilla Internetissä: <URL:http://docwiki.cisco.com/wiki/ONS_15454_Reference_Manual_R8.5.x_--_SNMP#SNMP_Overview>

/4/ The PHP Group. What is PHP? [Viitattu 12.2.2011] Saatavilla Internetissä: <URL:http://www.php.net/manual/en/intro-whatis.php>

/5/ Tobias Oetiker, OETIKER+PARTNER AG. rrdtool [Viitattu 13.2.2011] Saatavilla Internetissä: <URL:http://www.mrtg.org/rrdtool/doc/rrdtool.en.html>

/6/ The Apache Software Foundation. Apache HTTP Server Version 2.2 Documentation [Viitattu 13.2.2011] Saatavilla Internetissä: <URL: http://httpd.apache.org/docs/2.2/ >

/7/ Ubuntu documentation. Documentation for Ubuntu 8.04 LTS [Viitattu 1.2.2011] Saatavilla Internetissä: <URL: https://help.ubuntu.com/>

/8/ Ubuntu documentation. UpgradeNotes [Viitattu 1.2.2011] Saatavilla Internetissä: <URL:https://help.ubuntu.com/community/UpgradeNotes>

/9/ Nagios Enterprises LLC. Nagios - Nagios History [Viitattu 1.2.2011] Saatavilla Internetissä: <URL: http://www.nagios.org/about/history

/10/ GNU Operating System. GNU General Public License [Viitattu 1.2.2011] Saatavilla Internetissä: <URL:http://www.gnu.org/licenses/gpl.html>

/11/ Juergen Haas. libgd2 [Viitattu 18.2.2011] Saatavilla Internetissä: <URL:http://linux.about.com/cs/linux101/g/libgd2.htm>

/12/ Nagios Enterprises LLC. Nagios Core - CGI Configuration File Options [Viitattu 15.1.2011] Saatavilla Internetissä: <URL:http://nagios.sourceforge.net/docs/3_0/configcgi.html>

/13/ Nagios Enterprises LLC. Nagios Core - Configuration Overview [Viitattu 15.1.2011] Saatavilla Internetissä: <URL:http://nagios.sourceforge.net/docs/3_0/config.html>

/14/ Nagios Enterprises LLC. Nagios Core - About Nagios Core [Viitattu 13.1.2011] Saatavilla Internetissä: <URL:http://nagios.sourceforge.net/docs/3_0/config.html>

/15/ Net-SNMP. NEWS [Viitattu 4.5.2011] Saatavilla Internetissä: <URL:http://www.net-snmp.org/docs/NEWS.html>

/16/ Soren Dossing (2005), Alan Brenner, Ithaka Harbors (2008), Matthew Wall (2010). nagiosgraph README [Viitattu 5.4.2011] Saatavilla Internetissä: <URL:http://nagiosgraph.svn.sourceforge.net/viewvc/nagiosgraph/trunk/nagiosgraph/README)>

/17/ BalaBit IT Security Ltd. The syslog-ng Open Source Edition 3.2 Administrators Guide [Viitattu 5.4.2011] Saatavilla Internetissä: <URL:http://www.balabit.com/sites/default/files/documents/syslog-ng-ose-v3.2-guide-admin-en.html/index.html-single.html#id459748>

/18/ Ozeki Informatics Ltd. How to send SMS from Nagios [Viitattu 5.4.2011] Saatavilla Internetissä: <URL:http://www.sms-integration.com/p_110-nagios-sms.htm>

/19/ Nagios Enterprises LLC. Nagios Core – State Types [Viitattu 15.5.2011] Saatavilla Internetissä: <URL:http://nagios.sourceforge.net/docs/3_0/statetypes.html >

/20/ Nagios Enterprises LLC. Nagios Core – Distrinuted Monitoring [Viitattu 15.5.2011] Saatavilla Internetissä: <URL:http://nagios.sourceforge.net/docs/3_0/distributed.html>