LAUREA AMMATTIKORKEAKOULU Uuden edellä

Palvelinten hallinta SNMP-protokollan avulla

Malvisto, Antti

2011 Leppävaara

Laurea-ammattikorkeakoulu
Laurea Leppävaara

Palvelinten hallinta SNMP-protokollan avulla

Malvisto Antti
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Toukokuu, 2011

Laurea-ammattikorkeakoulu **Tiivistelmä**
Laurea Leppävaara
Tietojenkäsittelyn koulutusohjelma

Malvisto Antti

Palvelinten hallinta SNMP-protokollan avulla

Vuosi 2011 Sivumäärä 52

Tämä opinnäytetyö tehtiin käytännön projektina, jonka tarkoituksena oli keskittää yrityksen palvelinten valvonta yhden järjestelmän kautta toimivaksi. Tavoitteena oli saada suuri määrä informaatiota paremmin hallintaan, jotta pysyttäisiin tietoisina palvelinten resursseista, toimintakyvystä ja mahdollisista ongelmakohdista. Nykyiseen verkonhallintaan haluttiin saada lisää tehokkuutta automaattisen ympärivuorokautisen valvonnan avulla. Uusi järjestelmä mahdollistaa automaattisen valvonnan lisäksi palvelinten toimintojen ja toimintahistorian manuaalisen tarkkailun. Pysytään paremmin tietoisina siitä, mitä palvelimilla tapahtuu, milloin tapahtuu ja miksi tapahtuu. Tärkeänä lisänä palvelinten hallintaan haluttiin automaattiset hälytysviestit asianomaisten henkilöiden sähköpostiosoitteisiin ongelmatilanteiden sattuessa. Uusi järjestelmä kykenee mahdollistamaan edellä mainitun toiminnon.

Hanke toteutettiin Cacti-nimisellä verkonhallintaohjelmistolla. Cacti on avoimen lähdekoodin ohjelmisto eli se on täysin ilmainen ja vapaasti ladattavissa internetistä. Tavoitteena oli, että projektista ei aiheudu ylimääräisiä kustannuksia, joten tästä syystä avoin ohjelmisto oli oikea ratkaisu.

Projekti toteutettiin tv7-nimiselle yritykselle. Tv7 on koko Suomen kaapeliverkon kattava tv-kanava, joka näkyy tällä hetkellä myös osassa Viron kaapeliverkkoa. Voitaneen sanoa, että yrityksen aikaisemman toiminnan suhteen varsin laajassa mittakaavassa tv-tuotantoa nykyään tehdään ja siksi myös yrityksen sisäisen verkon hallinta on suuressa roolissa. Tässä työssä keskityttiin ainoastaan palvelinten valvontaan, mutta jatkokehityksen kannalta on tarkoitus lisätä uuteen verkonhallintajärjestelmään kaikki verkon laitteet ja työpöytäkoneet.

Asiasanat Verkonhallinta, Monitorointi, valvonta, SNMP

Laurea University of Applied Sciences **Abstract**
Laurea Leppävaara
Bachelor's Programme in Business Information Technology

Malvisto Antti

Server management with the Simple Network Management Protocol (SNMP)

Year 2011 Pages 52

This thesis was completed as a practical project for tv7, a TV program provider. The main objective was to centralize the server monitoring to be used only by one system. The Company's servers hold a great amount of information, which requires superior management in order to maintain the knowledge of the server resources, possible ongoing problems and the ability of the server to function correctly. The existing network management needed to acquire more efficiency with round-the-clock surveillance.

The objective was to remain more conscious of the incidents on the servers regarding the content of the incidents, when or why certain incidents occurred. The important addition enabled by the automatic surveillance was the automatic alerts sent to configured emails caused, for example, by a server down state.

The project was implemented with network management software called cacti. Cacti is open source software, which means that it is free of charge and downloadable from the Internet. The objective was to adopt a fully working management system without the expenses, which was the reason to choose the open source software.

Tv7 provides programs for the whole Finnish cable network and a part of the Estonian cable network as well. The scale of the company's operating area is wide and therefore the inner network management has a great part in the company's TV production. This project focuses on developing the management of the servers but it also supports the development of the whole company's network including all the network devices and the computers.

Keywords Network management, Monitoring, surveillance, SNMP

Sisällys

1 Johdanto

Tämän projektin tarkoituksena on mahdollistaa yrityksen käytössä olevien palvelinten keskitetty seuranta. Seuranta on tarkoitus toteuttaa SNMP-järjestelmän (Simple Network Management Protocol) avulla käyttäen siihen soveltuvaa hallintaohjelmistoa. Yritykselle uutena asiana palvelinten hallintaan tulee automaattinen monitorointijärjestelmä, joka seuraa palvelinten toimintaa ympäri vuorokauden ja raportoi verkon hallinnoijalle ei-toivotuista tapahtumista palvelimilla. Myös manuaalisen seurannan on tarkoitus helpottua, nopeutua ja selkeytyä. Palvelinten toiminnan ja suorituskyvyn seurannan on tarkoitus tapahtua yhden hallintajärjestelmän avulla. Yrityksellä on käytössä monta erillään toimivaa palvelinta ja näiden kaikkien seuranta yksilökohtaisesti vie paljon aikaa ja resursseja. Informaation määrä on hyvin suuri ja erikseen vaikeasti hallittavissa ja hahmotettavissa. Projektin tarkoituksena on tutkia, mahdollistaako keskitetty hallinta IT-henkilöstön ajalliset kustannussäästöt, jotta säästetty aika voitaisiin hyödyntää tehokkaammin muuhun työntekoon.

Automaatio mahdollistaa automaattiset hälytysviestit verkon ylläpitäjille, jos jotain tavallisuudesta poikkeavaa toimintaa tapahtuu palvelimilla. Nämä hälytykset on tarkoitus asettaa toimintakuntoon monitoroitaville palvelimille ja testata niiden toimivuus käytännössä. Hälytysviesteihin on teoriassa paljon nopeampi reagoida sen sijaan, että odoteltaisiin havaintoja tietoliikenteen ongelmista. Tavoitteena on tutkia, nopeuttavatko hälytykset todella reagointia ongelmatilanteissa. Verkonhallintaa on tarkoitus kehittää enemmän proaktiiviseen eli ennaltaehkäisevään suuntaan reagoivan toiminnan sijaan.

SNMP-järjestelmän avulla voidaan tallentaa statistiikkaa palvelinkoneilta ja muista SNMP-protokollaa tukevista verkon laitteista esimerkiksi kahden vuoden ajalta. Tämän tallennetun tiedon avulla saadaan graafisia tilastoja palvelinten komponenttien toiminnasta, mikä selkeyttänee komponenttien toiminnan seuraamista. Tallennettu tieto mahdollistaa myös verkon resurssien seurannan ja helpottaa verkon laajentamisen suunnittelua jatkossa. Verkon laajentaminen ei varsinaisesti ole osa tätä työtä, mutta se kuuluu projektin jatkokehitykseen, josta kerrotaan lisää raportin lopussa.

Yrityksellä ei ole aiemmin ollut käytössä automatisoitua verkonhallintajärjestelmää. Valvonta on tapahtunut ainoastaan manuaalisella tasolla sisäisen IT-henkilöstön toimesta. Hallintatekniikoina on käytetty ja käytetään edelleen kvm-kytkintä ja ssh-yhteyttä, joilla molemmilla voidaan ottaa myös etäyhteys palvelimiin. Edellä mainitut tekniikat ovat tärkeitä, niitä käytetään päivittäin, eikä niitä voida täysin korvata minkään tasoisella SNMP-järjestelmällä, koska ne vaativat sellaisten toimintojen suorittamista, joihin SNMP-järjestelmät eivät itsenäisesti kykene. SNMP-järjestelmä on vain lisäosa nykyiseen verkonhallintaan.

Yrityksen kehittämistarpeena on automatisoida kaikki sellaiset palvelinten valvontatoiminnot, jotka on mahdollista suorittaa automaattisella tasolla. Näitä toimintoja ovat palvelinten toimintakyvyn ja suorituskyvyn valvonta. Palvelinten toimintakyvyllä tarkoitetaan kaikkien komponenttien virheetöntä ja jatkuvaa toimintaa, ja suorituskyvyllä palvelinten kykyä suoriutua niille annetuista tehtävistä ilman, että mikään komponentti asettuu pullonkaulaksi ja vähentää siten palvelinkoneen tehokkuutta. Koska kyseessä on media-alan yritys, joka lähettää tv-ohjelmia muun muassa Suomen kaapeliverkkoon, mahdollisten ongelmatilanteiden ennakointi on erityisen tärkeässä asemassa, jotta vältyttäisiin lähetyskatkoilta ja muilta lähetykseen vaikuttavilta ongelmilta.

Tässä opinnäytetyössä tutkitaan myös, onko SNMP-järjestelmästä todellista hyötyä yritykselle vai onko vanha manuaalinen valvonta kokonaisuudessaan edelleen toimivampi ratkaisu. Voidaanko automaattisen järjestelmän avulla ennaltaehkäistä ongelmia ja tuleeko sen myötä uusia ongelmia? Voiko SNMP-järjestelmästä saatuun tietoon luottaa? Pohdinnan aiheena ovat myös automaattisen järjestelmän tuomat muut edut vanhaan järjestelmään nähden.

Yksi projektin kriteereistä oli toteuttaa se ilman varsinaisia investointeja, joten ohjelmistoksi valittiin työnantajan toimesta avoimen lähdekoodin ohjelmisto nimeltä Cacti. Cacti on varsin monipuolinen ohjelmisto ja internetissä olevasta cacti-yhteisöstä saa paljon tietoa ja dokumentaatioita ohjelmiston ja sen lisäosien käyttämisestä. Myöskään laitteiston puolesta ei ylimääräisiä kustannuksia kertynyt.

Tämä käytännön projekti tehtiin Taivas-tv7 nimiselle yritykselle. Tv7 on voittoa tavoittelematon media-alan yritys, joka lähettää 24 tuntia päivässä tv-ohjelmaa koko Suomen ja lähes koko Viron kattavaan kaapeliverkkoon. Ohjelmia voi seurata myös laajakaista-tv:n ja netti-tv:n kautta.

Yrityksen toimipisteet sijaitsevat Helsingissä ja Jerusalemissa Israelissa. Vakiohenkilökuntaa on noin 50 henkilöä. Vuonna 2009 Mediatum Oy:n tekemän tutkimuksen mukaan Tv7:n katsojamäärä Suomen kaapeliverkossa on viikoittain noin 191 000 henkilöä. Samana vuonna kanavan tunsi 28% suomalaisista. Kyseessä on kasvava yritys, joka laajentaa toimintaansa koko ajan. Tästä syystä hyvin suunniteltu verkonhallinta on tärkeä osa laajenemisprosessia ja tämän työn tarkoituksena on myös auttaa kyseisen prosessin kehittämisessä.

2 Verkonhallinta

Verkonhallinnalla tarkoitetaan verkon kautta saatavien palveluiden toimivuuden turvaamista. Ylläpitäjän tehtävänä on varmistaa, että palvelinten ja asiakkaiden väliset yhteydet toimivat, siirtokapasiteetti on riittävä ja yhteydet ovat luotettavia ja turvallisia. (Hakala & Vainio 2002, 269.)

Yksinkertaisuudessaan verkonhallinta on verkotettujen järjestelmien ja resurssien hallintaa, jotka yhdistävät nämä toisiinsa. Verkotetulla järjestelmällä voidaan tarkoittaa arkistoa, yhteyksiä ja tulostuspalvelimia yhdessä käyttäjien työasemien kanssa. Verkon resurssit voivat pitää sisällään reitittimiä, portteja, siltoja, kytkimiä, hubeja, modeemeja tai verkkokortteja ja viestejä, jotka yhdistävät nämä toisiinsa. (Simoneau 1999, 32.)

Verkko pitää sisällään seuraavat tekijät: käyttäjät ja heille määritellyt käyttöoikeudet, työasemien ohjelmistot, ohjelmistot jotka tarjoavat palveluja verkon resursseista ja laitteet, jotka tarjoavat pääsyn noihin resursseihin. (Simoneau 1999, 4.)

Paul Simeon mainitsee kirjassaan SNMP Network Management (1999,4) neljä pääsyytä verkonhallintaan, joita ovat ammatillisuus, taloudellisuus, teknisyys ja turvallisuus.

Ammatillisuus voi olla tietynlaista ylpeyttä omaa ammattiaan kohtaan, jolloin halutaan nostaa työpanoksen yleistä tasokkuutta. Hyvin hallittu verkko takaa yrityksen perusrakenteen, jota tarvitaan varmistamaan yrityksen tulevaisuus. (Simoneau 1999, 4.)

Taloudellisesti ajatellen verkonhallinta ei ole kallista. Se on työntekijöiden toimenkuvaan integroitavissa oleva toimi ja pääasiallisin tapa yritykselle tuottaa enemmän tuotteita ja palveluita alhaisempaan hintaan. Hyvin hallitun verkon ylläpito maksaa yritykselle korkeintaan 25 % kuluista. Kun verkkoa hallitaan tehokkaasti ja kaikki sen osat on dokumentoitu, on paljon nopeampaa paikantaa ongelmat ja havaita kehityksen tarpeessa olevat osa-alueet. Hyvin hallittu verkko lisää siis tuottavuutta, koska verkon ennakoiva hallinta vähentää verkon romahtamisten ajallista kestoa. Verkon toimivuus koskee kaikkia niitä henkilöitä, jotka tarvitsevat verkkoa työskennelläkseen. (Simoneau 1999, 4.)

Tekniseltä kannalta hyvin hallittu verkko on yritykselle melkoisen tärkeä. Itsenäisesti toimimaan jätetty verkko voi tarjota jonkinlaista palvelun tasoa, mutta se ei kykene tarjoamaan lisäpalveluita, lisäämään uusia käyttäjiä, päivittämään nopeusluokkia tai kehittämään toimintavarmuutta, kuten hyvin hallittu verkko. Itsenäisesti toimiva verkko ei myöskään kykene mukautumaan yrityksen muuttuviin tavoitteisiin. (Simoneau 1999, 4.)

Turvallisuuden kannalta perusteellisesti dokumentoitu hyvin hallittu verkko on helpompi suojata, kuin itsenäisesti toimiva verkko. Tunnetut ja tuntemattomat käyttäjät, laitteet ja verkon resurssit on helpompaa tunnistaa ja varmentaa, että palveluja tarjotaan vain valtuutetuille käyttäjille. Esimerkiksi vasteaika verkkoon tunkeutujan löytämiseksi on paljon lyhyempi hyvin hallitussa verkossa. (Simoneau 1999, 4.)

Verkonhallinnan päätavoitteeksi voitaisiin tiivistää käyttäjän mahdollisuus päästä käsiksi yrityksen verkkoresursseihin mihin aikaan tahansa, olipa kyse sitten sisäverkosta tai yhteyksistä internetiin. (Simoneau 1999, 6.)

Tv7:n verkonhallinnan tavoitteina on työasemien, palvelinten ja verkkolaitteiden toimivuuden takaaminen ja vikojen nopea paikallistaminen ja niiden ennakointi. Yritykselle asennettavan automaattisen valvontajärjestelmän tulisi edistää vikojen nopeaa korjausta.

Palvelinten hallintaan liittyviä tekniikoita on paljon ja seuraavaksi käydään läpi kolme erilaista tekniikkaa niiden hallinnointiin. Näitä tekniikoita ovat SSH-yhteys, kvm-kytkin ja SNMP, joista yrityksellä on jo käytössä SSH ja kvm. Tarkoituksena on vertailla näiden hyötyjä ja haittoja toisiinsa nähden. Tarkempi analyysi löytyy raportin yhteenvedosta. Ennen näitä tekniikoita käydään vielä läpi osa-alueita, joista verkonhallinta kokonaisuudessaan koostuu.

2.1 Asetusten hallinta

Asetusten hallinta on yleensä ensimmäinen asia, jonka kanssa verkon ylläpitäjä joutuu tekemisiin. Se alkaa ensimmäisen verkkoasetuksen tekemisestä, eikä lopu koskaan. Jotta verkonhallinta olisi jatkossa toimivaa, on syytä dokumentoida alusta lähtien kaikki verkon asetukset, sillä mitä tarkemmin ne on dokumentoitu, sitä vaivattomampaa verkonhallinta on. Kaksi suurinta haastetta ovat dokumentaation työstäminen käytettävään muotoon ja tiedon sopivuus. Jotta tieto olisi parhaiten saatavilla, kun sitä eniten tarvitaan, se tulisi järjestää kahteen muotoon: toiminnallisuuden mukaan ja rakenteen mukaan. Dokumentaatiossa tulisi myös erottua toisistaan kaksi alakategoriaa: laitteisto ja ohjelmistot. Laitteiston määrittäminen on hyvin lähellä inventaarion tekemistä. Sen tulisi sisältää tarkasti kaikki tiedot nimeämisistä kaapelointien pituuksiin, jokaisen järjestelmän sijainnin verkossa ja kaikkien laitteiden valmistajat. (Simoneau 1999, 268.)

Ohjelmiston asetusten määrittely on monimutkaisempaa, koska ne muuttuvat koko ajan. Haastavaa siitä tekee myös se, että ohjelmistoasetukset ovat suhteessa toisiin asetuksiin ja laitteistoon, jonka päällä niitä ajetaan. Työasemien ja palvelinten väliset suhteet ovat tärkeitä, mutta yhtä tärkeitä ovat työasemien keskinäiset suhteet. Myös työasemien ja reitittimien,

palvelinten ja reitittimien ja reitittimien välisiä suhteita tulee miettiä. Ohjelmiston dokumentaation tulisi keskittyä enemmän toiminnallisuuksien näkökantoihin. (Simoneau 1999, 268.)

Verkonhallinnoijien tulee varmistua siitä, että he ovat tietoisia muuttuvista asetuksista voidakseen optimoida verkon toimivuutta. (Simoneau 1999, 268.)

2.2 Laskennan hallinta

Laskennan hallinta pitää sisällään käyttötarkastuksen omaisuuden hallintaa käyttäen ja verkon arvon arvioinnin.

Omaisuuden hallinnan avulla pidetään kirjaa siitä, mitä kaikkea verkko sisältää. Tästä esimerkkinä verkkokortit, jotka ovat korvattavissa, työasemat joiden paikkaa voidaan vaihdella, kaapelointi ja niihin liittyvät laitteet, palvelimet, siltaavat laitteet, kytkimet ja reitittimet. Kaikki muutokset laitteistossa tulisi dokumentoida. (Simoneau 1999, 269.)

Käyttötarkastuksella tarkoitetaan selontekoa siitä, kuka käyttää verkon resursseja ja mitä noilla resursseilla tehdään. Tämä ei eroa kovinkaan paljon resurssien kulunvalvonnan tarkastelusta tietoturvallisuuden näkökannasta katsottuna. Verkossa olevien järjestelmien arvo tulee olla arvioituna mahdollisten ongelmatilanteiden ja laitteiston korvaamisen varalta. Näiden ilmoitettujen arvojen tulisi osoittaa jokaisen järjestelmän tärkeys yrityksen tavoitteiden saavuttamiseksi. (Simoneau 1999, 269.)

2.3 Suorituskyvyn hallinta

Suorituskyvyn hallinnan avulla voidaan vaikuttaa verkon resurssien luotettavuuteen, saatavuuteen ja suorituskykyyn. Suorituskyvyn hallinta pitää sisällään monenlaisia tehtäviä, joita ovat mm. perusteiden määrittely, kehityssuunnan analysointi, tiedotusvälineiden testaus, simulointi, mallinnus, suunnittelu ja uudelleen soveltaminen. (Simoneau 1999, 269,270.)

Perusteiden määrittäminen tai verkon toiminnan testaaminen tietyllä ajanjaksolla sen perustason selvittämiseksi, on ensimmäinen tehtävä. Perustason tulisi olla keskimääräinen kuorma (pakettia/sekunti) tietyllä ajanjaksolla verkkosegmenttiä kohden. Koska verkkoa säädetään optimaalisen suorituskyvyn saavuttamiseksi, on suorituskyvyn seuranta tärkeää. Sen avulla verkon ylläpitäjä voi havaita ja analysoida verkon kausiluontoisia suuntauksia ja tehdä esimerkiksi kuukausittaisia ja viikoittaisia vertailuja. (Simoneau 1999, 269,270.)

Suorituskyvyn ja luotettavuuden valvonta voi sisältää esimerkiksi pirstoutumisen onnistumisia ja epäonnistumisia, uudelleenlähetyksiä, ristiriitoja, ping-viiveen kierrosaikoja ja virheitä.

Kun kaikki nämä suorituskykyyn vaikuttavat tekijät ovat kohdallaan, on helpompaa suunnitella verkon laajentamista. Kun perustaso verkon kuormitukselle ja muille tekijöille on määritelty, voidaan arvioida tulevaisuuden tarpeita käyttäen simulointi- ja mallinnusohjelmistoja. (Simoneau 1999, 269,270.)

Muita suorituskykyyn liittyviä huolenaiheita ovat verkon saatavuus ja verkkolaitteiden ruuhkautuminen. Verkon saatavuuden turvaamiseksi on syytä valvoa palvelimia, jotta voidaan turvata yhteydet TCP-sovelluksia varten. Reitittimien, kytkinten ja siltaavien laitteiden ruuhkautumisten paljastamiseksi pakettien suodatus on hyvä indikaattori. Verkonvalvojat voivat myös tarkastella virheettömiä datagrammeja, joita järjestelmät heittävät pois tehdäkseen lisää tilaa sisään tulevaa, ulospäin menevää tai reititystä varten olevaa puskurimuistia. Joskus tämä on niinkin yksinkertaista, kuin verkkoliikenteen ulospäin menevän jonon tarkastelua. Verkon saatavuuden ongelmat ovat vahvasti liitoksissa turvallisuuden ja virheiden hallintaan. (Simoneau 1999, 269,270.)

2.4 Turvallisuuden hallinta

Turvallisuuden hallinta on todennetun tiedon pitämistä salassa niiltä tahoilta, joilla ei ole siihen oikeuksia. Verkonhallinnan tavoitteet esimerkiksi turvallisuuden suhteen eivät kuitenkaan saisi kärsiä verkonkäyttäjien vapaasta kulusta verkossa. (Simoneau 1999, 270,271.)

Turvallisuuden hallinta ryhmittyy neljään kategoriaan: datan arvon arviointiin, riskien arviointiin, tarkastuksiin ja arkiston hallintaan. Joskus kaikki varastoitu verkon tieto tulee arvioida, jotta voidaan miettiä investointeja tiedonturvaamiseen jatkossa. Sen sijaan, että datan arvon arvioinnilla tarkoitettaisiin jokaisen tiedoston tarkistamista, se on enemmänkin tiedostotyyppien tarkkailua. Kun tiedetään millaista tietoa halutaan suojella, on aika arvioida riskit. Esimerkiksi kuinka moni käyttäjä voi kirjautua useampaan, kuin oman jaoston verkkoon? Käyttävätkö nämä moninaisiin jaostoihin kuuluvat käyttäjät samaa salasanaa kirjauduttaessa eri jaostojen palveluihin? Onko verkosta pääsy internettiin? Kuka voi kirjautua palvelimille, jotka sisältävät kriittistä dataa? Nämä kysymykset herättävät kysymyksiä muun muassa fyysisen kulunvalvonnan ongelmista. Onko valtuuttamattomilla henkilöillä pääsy järjestelmään, josta pääsee käsiksi kriittiseen dataan tai miten verkon hallinnoija tietää jos verkkoon on päässyt tunkeutuja etäyhteyden avulla? (Simoneau 1999, 271.)

Turvallisuustarkastuksissa on syytä miettiä edellä mainittuja kysymyksiä. Näissä tarkastuksissa ei ainoastaan tarkkailla palvelimilla tapahtuvaa liikennettä, vaan etsitään myös mahdollisia ohjelmistoja, jotka puolestaan etsivät järjestelmien tietoturva-aukkoja.

Ohjelmistojen sisäänkäyntitietoja tulisi myös tarkkailla. Näitä tietoja ovat esimerkiksi salasanojen kokeilut sisältäen ajankohdan, epäonnistuneiden yritysten määrän ja epäonnistuneen salasanan käyttö jokaisella yrityskerralla. Mikä palvelin suorittaa käyttäjän sisäänpääsyn online-tilassa? Mitä hallinnollisia komentoja kirjoitettiin, milloin ja kenen toimesta? Onko olemassa monta eri lokia kaikkien käyttäjien toimista verkossa? Onko palveluaiheisia viestejä esimerkiksi kopioinnoista, varmuuskopioista tai suoritetuista tulosteista? (Simoneau 1999, 271,272.)

Kun muun muassa näihin kysymyksiin on saatu vastaus, täytyy tieto niistä tallentaa varmaan paikkaan, josta se on helposti saatavilla, kun sitä tarvitaan. Tähän tietoon on määritelty verkon turvallisuuden perusteet ja sen avulla on paljon nopeampaa ja helpompaa napata häiriötekijöitä. (Simoneau 1999, 272.)

2.5 Virheiden hallinta

Virheiden hallinnalla pyritään ennakoimaan virheitä ja korjaamaan niitä, joita on mahdoton ennustaa. Tämä on mahdollista jos aiemmin mainitut asetusten, laskennan, suorituskyvyn ja turvallisuuden hallinnan tiedot ovat saatavilla. Virheitä voidaan havaita esimerkiksi seuraavilla tavoilla:

- Havainnot siitä, että nykyiset arvot eivät täsmää perusarvojen kanssa.
- Hallittavat laitteet lähettävät virheviestejä.
- Verkon käyttäjiltä tulee suoraa palautetta joko helpdeskiin tai verkon ylläpitäjälle. (Simoneau 1999, 272.)

Ongelman löydyttyä sitä aletaan analysoida ja mietitään mikä on todellinen ongelma oireiden sijaan. Kun ongelma on selvitetty, on tärkeää kirjata ylös ratkaisut, jotka auttoivat ongelmaan ja ratkaisut, jotka eivät auttaneet. Tällä tavoin virheiden hallinta on jatkossa toimivampaa. (Simoneau 1999, 272,273.)

2.6 Verkonhallintatekniikat

Uuden SNMP-järjestelmän myötä aiemmin verkonhallinnassa käytetyt tekniikat eivät ole poistuneet käytöstä. Ne ovat edelleen tarpeellisia, hyödyllisiä ja tehokkaita tapoja palvelinten hallinnoimiseen. Yrityksellä on käytössä SSH-yhteydet palvelimiin ja kvm-kytkin, jotka molemmat mahdollistavat tarpeelliset etäyhteydet.

2.6.1 SSH

Secure Shell (SSH) on suomalaisen Tatu Ylösen kehittämä menetelmä, jolla saadaan TCP/IP-protokollan avulla yhteys järjestelmästä toiseen järjestelmään. Komentoja ja niiden tulosteita voi turvallisesti katsella Secure Shellin sisältämän komentotulkin avulla. SSH on verrattavissa toiminnallisuuksiltaan aikaisempaan Telnet-palveluun, mutta SSH:n käyttäjien väliset viestit lähetetään salattuina, joten ulkopuolisten henkilöiden, esimerkiksi hakkereiden on vaikeaa päästä käsiksi yksityisiin tietoihin, kuten käyttäjätunnuksiin tai salasanoihin. (McCarty 2005, 251.)

SSH tarjoaa myös muita natiiveja palveluita, kuten tiedoston kopioinnin ja etäkäskyjen toteuttamisen. SSH-tunnelointi on myös mahdollinen toiminto. Siinä muodostetaan suojattu SSH-istunto, jossa satunnaiset TCP-yhteydet tunneloidaan verkon läpi. Tunneleita voidaan luoda milloin tahansa, eikä siihen tarvitse käyttää paljon aikaa. (SSH Port Forwarding 2010.)

SSH:sta on olemassa kaksi versiota, joita ovat SSH 1 ja SSH 2. SSH 1 -versiossa asemille on annettu RSA-avaimet (Rivest Shamir Aldeman), joiden avulla ne tunnistetaan. Palvelimelta löytyy palvelinavain, joka vaihtuu määräajoin. Työasema lähettää pyynnön palvelimelle ja saatuaan siltä julkiset asema- ja palvelinavaimet, työasema vertaa niitä omaan tietokantaansa. Tämän jälkeen muodostetaan istunnon salausavain siten, että työasema muodostaa satunnaisluvun, jonka myös palvelin kykenee tulkitsemaan. Palvelin ehdottaa käytettäväksi salausalgoritmeja, joista työasema valitsee haluamansa. Kaikki liikenne on salattua. Asematunnistuksen ja salausmenetelmän valinnan jälkeen on vuorossa kirjautuminen ja käyttäjän tunnistus. (Puska 2001, 267.)

2.6.2 SSH 1

SSH 1 sisältää seuraavat ominaisuudet: rhosts, jolla tarkoitetaan kirjautumista ilman käyttäjän tunnistamista. Rhosts Authentication, jolla tarkoitetaan käyttäjän varmentamista sisäänkirjautumisen yhteydessä. RSA on julkisen avaimen salausalgoritmi, jota käytetää RSA Authentication -todennuksessa. Password Authentication tarkoittaa salasanalla tunnistautumista.

Rhosts: SSH-palvelin myöntää kirjautumisoikeuden ilman salasanatunnistusta, jos työasema on määritelty turvalliseksi palvelimella ja käyttäjätunnus on molemmissa koneissa sama. Tätä menetelmää ei suositella käytettäväksi sen heikon tietoturvatason vuoksi. (Puska 2001, 267.)

Rhosts Authentication: Sisäänkirjautuminen sallitaan vain, jos työasema on määritelty turvalliseksi palvelimella, se löytyy SSH-palvelimen listasta ja RSA-avain voidaan varmentaa. (Puska 2001, 267-268.)

RSA Authentication: Käyttäjä luo julkisen ja yksityisen salausavaimen. Jos palvelin sisältää kopion julkisesta avaimesta, se lähettää satunnaisen salatun haasteen julkisella avaimella, jonka työasema voi purkaa käyttäjän salaisella avaimella. Tunnistuksen jälkeen kirjautumiseen ei enää vaadita salasanaa. (Puska 2001, 268.)

Password Authentication: Palvelin hyväksyy yhteyspyynnön vain tunnistamaltaan käyttäjältä, jonka salasanaa se vertaa itse laskemaansa ja joko hyväksyy tai hylkää pyynnön. (Puska 2001, 268.)

2.6.3 SSH 2

Tässä SSH versiossa Diffie-Hellman-istuntoavain korvaa aiemmin käytetyt palvelinavaimet, mikä mahdollistaa vahvempien salausmenetelmien käytön istuntojen salaukseen. Luotettaviin asemiin pohjautuvaa käyttäjätunnistusta ei tueta ollenkaan SSH 2 versiossa, vaan mahdollisia ovat ainoastaan seuraavat menetelmät:

DSA Authentication: RSA-algoritmin sijaan käytetään DSA-salausalgoritmia, joka pitää sisällään julkisen ja yksityisen avaimen. Palvelin purkaa työaseman yksityisellä avaimella salaaman istuntotunnuksen työaseman julkisella avaimella. Istuntotunnuksen tuntevat ainoastaan palvelin ja työasema. (Puska 2001, 268.)

Password Authentication: Käyttäjätunnus ja salasana lähetetään salattuna ja palvelin tunnistaa käyttäjän. (Puska 2001, 268.)

Yrityksessä käytetään SSH 2 versiota, koska siinä on vahvemmat salausalgoritmit ja luotettaviin asemiin perustuvaa käyttäjätunnistusta ei käytetä ollenkaan. Toisin sanoen SSH:n versio 2 on tietoturvallisempi ratkaisu.

Yrityksessä SSH-yhteyttä käytetään Linux-palvelimiin. Koska SSH sisältää komentotulkin, sen avulla voidaan suorittaa mitä tahansa toimintoja. Esimerkkinä tällaisia toimintoja ovat mm. verkkoliikenteen seuranta, kiintolevyjen tilan valvominen ja käyttäjien käyttöoikeuksien määritykset jne.

Lähetyspalvelimiin SSH:lla ei saa yhteyttä, koska niillä ajetaan Windows-järjestelmän päällä suoritettavia ohjelmistoja, joita on mahdotonta hallita komentorivin kautta.

2.6.4 Kvm-kytkin

Kvm-kytkin mahdollistaa usean tietokoneen käytön yhdellä näytöllä, näppäimistöllä ja hiirellä. Halutut tietokoneet liitetään samaan kvm-kytkimeen, jonka avulla voidaan hallita näitä koneita kutakin vuorollaan.

Kvm-kytkin säästää huomattavasti tilaa ja se vähentää kustannuksia, kun ei tarvitse hankkia näyttöä, näppäimistöä ja hiirtä jokaiselle koneelle. (Planet KVM 2010)

Yrityksen käytössä olevan kytkimen mukana toimitettiin ohjelmisto, joka mahdollistaa tietokoneiden etäkäytön sallituista MAC-osoitteista (Media Access Control). Kytkimeen on liitettynä Windows ja Linux palvelinkoneita, joihin saadaan muodostettua nopeasti yhteydet.

2.7 SNMP-protokolla

SNMP-protokolla (Simple Network Management Protocol) on sovelluskerroksen protokolla, joka helpottaa hallintatietojen vaihtoa eri verkkolaitteiden välillä. Se on osa TCP/IP-protokollasarjaa (Transmission Control Protocol/Internet Protocol). (Simple Network Management Protocol 2011.)

SNMP päättää pyydetyn datatyypin perusteella millaisen viestin se lähettää agentille ja millaisia viestejä se ottaa niiltä vastaan. SNMP-protokolla voi toimia myös muiden, kuin TCP/IP-protokollan päällä. Peruskokoonpanossaan SNMP käyttää TCP/IP-protokollasarjaa (erityisesti UDP:tä,IP:tä ja ethernettiä) järjestääkseen yhteydet hallittavien IP-osoitteilla varustettujen laitteiden kanssa. (Simoneau 1999, 35.)

SNMP-protokolla mahdollistaa verkon suorituskyvyn valvomisen, verkkoon liittyvien ongelmien paikallistamisen ja korjaamisen ja suunnittelun verkon laajentamiseksi. (Simple Network Management Protocol 2011.)

SNMP-järjestelmä mahdollistaa tietoverkon kehittämisen, koska sen avulla voidaan paljastaa verkon heikot kohdat ja muokata niitä toimivammiksi.

SNMP-protokollasta puhuttaessa valvonnan painopiste on laitteistovirheiden automaattisessa havaitsemisessa ja komponenttien kuormituslaskennassa. Valvontatoimintojen tietoturvallisuus on melko vähäistä luokkaa ja turva-asetusten määrittely pienissä ja keskisuurissa yrityksissä kannattaa keskittää esim. reitittimien huolelliseen konfigurointiin tietoturvaohjelmistojen hankinnan sijaan. Huolellisella konfiguroinnilla voidaan estää 80 – 90 prosenttisesti potentiaaliset tunkeutumisyritykset. Ottamalla käyttöön salausmenetelmät, jotka sisältävät käyttä-

jien laajan tunnistuksen (authentication) voidaan saavuttaa tietoliikenteen osalta lähes täydellinen tietoturvallisuus. (Hakala & Vainio 2002, 269.)

SNMP-protokollasta on olemassa kolme versiota, joita ovat SNMP versio 1 (SNMPv1), SNMP versio 2 (SNMPv2) ja SNMP versio 3 (SNMPv3).
Yrityksellä on käytössä lukuisa määrä verkkoon liitettäviä laitteita, jotka tukevat SNMP-protokollaa ja näin ollen mahdollistavat niiden automaattisen monitoroinnin. SNMP-järjestelmään liitettyjä laitteita voidaan tarkkailla palvelimelle asennetulla selain-pohjaisella ohjelmistolla, joka tässä tapauksessa on Cacti niminen ohjelmisto.

SNMP-protokollan tulee pitää kirjaa ja korjata ristiriitaisuuksia hallittavien laitteiden välillä. Erilaiset tietokoneet käyttävät erilaisia datan esitystekniikoita, jotka voivat vaarantaa SNMP:n kyvyn vaihtaa informaatiota eri laitteiden välillä. SNMP käyttää ASN.1:stä (subset of Abstract Syntax Notation One) mukauttaakseen tiedon välitystä vaihtelevien järjestelmien välillä. (Simple Network Management Protocol 2011.)

Tässä työssä ei keskitytä niinkään snmp-protokollan tietoturvallisuuteen, vaan järjestelmän luotettavaan toimivuuteen ja palvelinten monitorointiin. Ennen kuin käydään läpi verkonhallintaohjelmisto cactia ja sen ominaisuuksia, luodaan katsaus itse SNMP-protokollaan.

2.8 SNMP:n historia

SNMP protokolla otettiin käyttöön vuonna 1988. Protokollan laatijat olivat Jeffrey Case, Mark Fedor, Marty Schoffstall ja James Davin. SNMP-prosessi aloitettiin edellä mainittujen henkilöiden toimesta jo vuonna 1987, jolloin kehitettiin SNMP:n edeltäjä SGMP (Simple Gateway Monitoring Protocol). Määrittelynimeksi ehdotettiin RFC1028:aa. Vaikka SGMP tarkoitettiin pääosin reitittimien hallintaan, sitä käytettiin myös muiden järjestelmien monitorointiin. Samaan aikaan kun SGMP RFC1028 julkaistiin, ilmestyi kaksi muunlaista verkonhallintaratkaisua. Syntyi kaksi erilaista dokumentaatiota: High-Level Entity Management System (HEMS) ja Common Management Information Protocol (CMIP). (Simoneau 1999, 8.)

IAB:n (Internet Architecture Board) kokouksessa vuonna 1988 toimintaperiaatteeksi määrättiin RFC1052. SNMP olisi väliaikainen verkonhallintatyökalu ja CMIP olisi pitkäaikainen ratkaisu. (Simoneau 1999, 8.)

RFC1065 esitti SMI:n (Structure of Management Information) perustuen ISO:n (International Organization for Standardization) ohjelmoimaan ASN.1 (Abstract Syntax Notation One) rajapintaan. RFC1066 määritteli yksityiskohtaisesti ensimmäisen MIB:n (Management Information

Base) hallinnoimaan TCP/IP (Transmission Control Protocol / Internet Protocol)-pohjaisia verkkoja. (Simoneau 1999, 8.)

SNMP RFC1067 määriteltiin vuonna 1989 SNMP:n ensimmäiseksi versioksi (SNMPv1) ja SMIv1 määriteltiin käytettäväksi samaan aikaan. Samana vuonna yli 20 kauppiasta esitteli heidän SNMP ratkaisujaan. 1990-luvulla kauppiaiden määrä nousi yli 120:een, joka johtui määrittelyjen paranteluista. 1990-luvulla SNMPv1 käytti jo RFC-versiota 1157. (Simoneau 1999, 8-9.)

Vuonna 1992 Internet Engineering Task Force (IETF) muodosti työryhmän SNMPv2:n suunnitteluun. SNMPv2 otettiin virallisesti käyttöön vuonna 1993. Vuonna 1995 otettiin käyttöön kehitelty yhteisöpohjainen versio SNMPv2:sta nimeltään SNMPv2c. (Simoneau 1999, 10,12.)

SNMPv3:n työryhmä perustettiin vuonna 1997 IETF:n toimesta. Tarkoituksena oli jatkaa SNMPv2: kehittelyä ja määrittää standardit turvallisuudelle ja hallinnalle. Varsinainen SNMPv3 julkaistiin vuonna 1998 ja sen parantelua jatkettiin heti vuonna 1999 julkaisemalla ainakin kaksi lisämäärittelydokumenttia. (Simoneau 1999, 14-16.)

2.9 SNMP:n peruskomponentit

SNMP:n avulla toteutettu verkko koostuu kolmesta osasta, joita ovat hallinta-asema, hallinta-agentti ja hallittavat laitteet. (Hakala & Vainio 2002, 269.)

Tässä hankkeessa hallinta-asemana toimi yksi palvelin ja hallittavina laitteina muutama tuotantokäytössä oleva palvelin.

2.9.1 Hallittavat laitteet

Hallittavilla laitteilla tarkoitetaan hallintaverkkoon liitettyjä laitteita, jotka sisältävät SNMP-agentin. Nämä laitteet keräävät ja varastoivat tietoja SNMP-protokollan avulla hallinta-aseman käyttöön. Hallittavia laitteita voivat olla esimerkiksi reitittimet, palvelimet, kytkimet, siltaavat laitteet, hubit, tulostimet ja kaikki laitteet, jotka tukevat SNMP-protokollaa. (Hakala & Vainio 2002, 269.)

2.9.2 Hallinta-asema

Hallinta-asema sisältää ohjelmistoja, joiden avulla hallitaan ja valvotaan verkon laitteita. Hallinta-asemat tarjoavat suurimman osan resursseista prosessointia ja muistia varten, joita tarvitaan verkon hallintaan. Yhtä hallittavaa verkkoa kohden täytyy olla vähintään yksi hallinta-asema. (Hakala & Vainio 2002, 270.)

Hallinta-asemaksi voidaan valita mikä tahansa kone. Hallintakonetta täytyy pitää käynnissä ympäri vuorikauden ja siksi olisi hyvä säilyttää sitä jäähdytetyssä tilassa tai vähintäänkin varmistaa, että koneen komponentit ovat hyvin jäähdytetyt ja lämpötilat pysyvät kurissa. Kohdeyrityksen hallinta-asemana toimii yksi palvelimista, joka on sijoiteltu ilmastoituun palvelinhuoneeseen.

2.9.3 Hallinta-agentti

Jokainen SNMP-protokollaa tukeva laite sisältää oman hallinta-agentin, joka kerää tietoa laitteen toiminnasta. Hallinta-agentti joko odottaa kyselyä laitteen toiminnasta hallinta-asemalta tai se voi itsenäisesti lähettää sille ns. trap-sanomia. Näitä viestejä varten agenttiohjelmistossa on määritelty halutuille tapahtumille tietty raja-arvo, joka ylittyessään lähettää tiedon arvon ylittymisestä hallinta-asemalle. (Hakala & Vainio 2002, 270.)

Raja-arvoja on olemassa suuri määrä, joita ovat esimerkiksi prosessien määrän ylittyminen, prosessorin ylikuormittuminen, kiintolevyn levytilan täyttyminen tai verkkoliikenteen ylikuormittuminen.

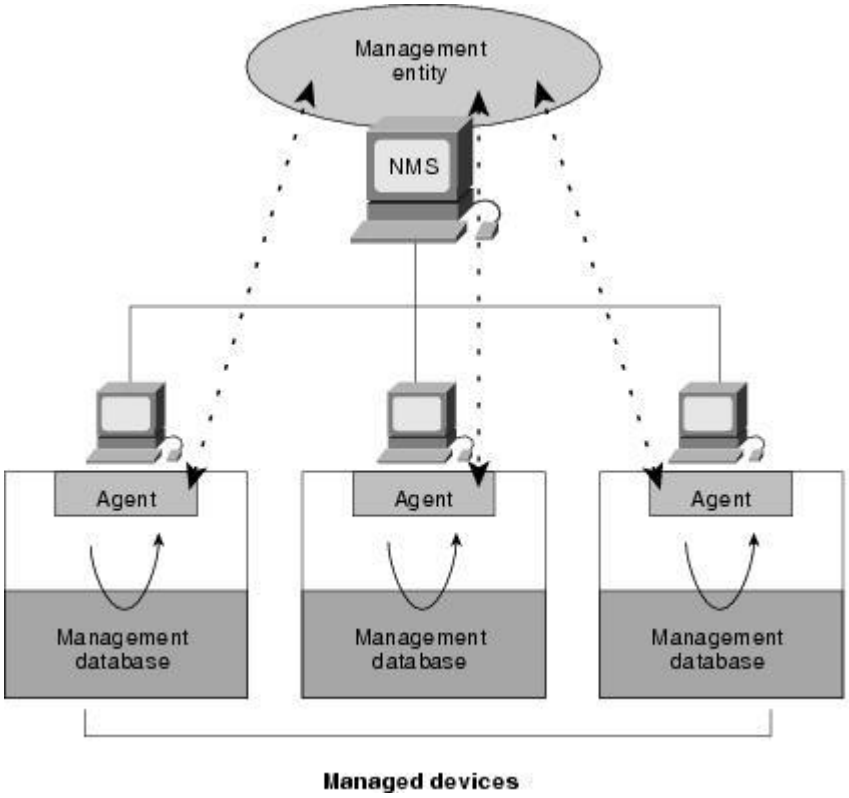
Saatuaan trap-viestin hallinta-agentilta, hallinta-asema tekee seuraavaksi mitä sen asetuksiin on määritelty. Se voi esimerkiksi lähettää sähköposti-ilmoituksen verkon hallinnoijalle tietyn tapahtuman raja-arvon ylittymisestä, jolloin hallinnoija voi nopeasti reagoida tilanteeseen heti luettuaan sähköpostiviestin.

2.9.4 Hallintatietokanta

Hallintatietokanta on virtuaalinen tietokanta, joka tunnistaa jokaisen hallittavan objektin sen nimen, syntaksin, saatavuuden, tilan, tekstikuvauksen ja ainutlaatuisen OID (Object Identifier) numeron perusteella. (Simoneau 1999, 33.)

MIB-tietokanta (Management Information Base) sisältää kaikista laitteista kerätyt tiedot, niiden tietotyypit sekä muuttujat, joilla hallinta-asema vaikuttaa laitteiden toimintaan. Tietokanta koostuu osittain pakollisista ryhmitellyistä objekteista, jotka on standardoitu puumaiseksi rakenteeksi. Nämä objektit löytyvät kaikista SNMP-järjestelmään lisätyistä laitteista ja niihin viitataan tunnisteilla (Object Identifier, OID), jotka kertovat niiden sijainnin puurakenteisessa tietokannassa. Hallinta-asema lähettää oman MIB-tietokantansa sisältämien objektien perusteella kyselyn agentille, joka palauttaa vastaavan objektin, jos sellainen löytyy sen omasta tietokannasta. Ilman vastaavaa objektia tulee virheilmoitus. (Hakala & Vainio 2002, 270.)

Ciscon dokumentaatiosta poimittu Kuva 2.9.4.1 havainnollistaa hyvin SNMP-protokollan rakenteen ja sen eri osien suhteet toisiinsa nähden. (Simple Network Management Protocol 2011.)



Kuvio 1. SNMP-protokollaa käyttävät laitteet.

Oikein toimiakseen hallinta-aseman ja agentin täytyy käyttää johdonmukaisesti samaa hallintatietokantaa (MIB). Koska hallinta-asemalla ja hallittavan laitteen sisältävällä agentilla on molemmilla olemassa oma tietokanta, hallinta-aseman hallinnoijan tulee laatia tarkka kopio jokaisen hallinta-agentin hallintatietokannasta hallinta-asemalle. Hallinta-aseman pyytäessä dataa agentilta sen täytyy kysyä tunnistetta (OID), jonka agentti voi tunnistaa. (Simoneau 1999, 34.)

2.10 SNMP-komennot

Hallittavia laitteita monitoroidaan ja ohjataan neljällä peruskomennolla: **luku**, **kirjoitus**, **trappi** ja **suunnatut toimenpiteet**. **Luku**-komentoa käytetään hallinta-aseman toimesta valvomaan hallittavia laitteita. Hallinta-asema tutkii eri muuttujia, jotka ovat hallittavien laitteiden ylläpitämiä. Hallinta-asema käyttää **kirjoitus**-komentoa hallitakseen laitteita ja se myös muuttaa hallittujen laitteiden sisältämiä muuttujien arvoja. Hallittavat laitteet käyttävät **trappi**-komentoa epäsynkronoidusti raportoidakseen tapahtumista hallinta-asemalle. Kun

tietyn tyyppisiä tapahtumia ilmenee, laitteet lähettävät informatiivisia viestejä hallinta-asemalle. (Simple Network Management Protocol 2011.)

Hallinta-asema käyttää **suunnattuja toimenpiteitä** määritelläkseen mitä muuttujia hallittava laite tukee. Se myös kerää järjestelmällisesti tietoa muuttujatauluista, kuten reititystauluista. (Simple Network Management Protocol 2011.)

2.11 SNMP-versio 1

SNMPv1 on SNMP-protokollan alkuperäinen toteutusversio. Se toimii muun muassa UDP- (User Datagram Protocol), IP- (Internet Protocol), CLNS- (Connectionless Network Service), DDP- (Apple Talk Datagram-Delivery Protocol) ja IPX-protokollilla (Novell Internet Packet Exchange). (Simple Network Management Protocol 2011.)

Hallintatiedon rakenne (SMI) määrittelee säännöt hallintatiedon kuvaamiseen käyttäen merkkikieltä Abstract Syntax Notation One (ASN.1). SMI tekee kolme avainmäärittelyä: ASN.1 datatyypit, tarkoin määriteltyjä SMI-datatyyppejä ja SNMP hallintatietokantatauluja. (Simple Network Management Protocol 2011.)

2.11.1 SNMP-versio 1 and ASN.1 Datatyypit

SNMPv1:n SMI määrittelee, että kaikilla hallittavilla kohteilla on ASN.1:n tietty yhteenkuuluva osajoukko. Kolme ASN.1:n datatyyppiä vaaditaan: nimi, syntaksi ja koodaus.

- Nimi toimii objektin tunnisteena.
- Syntaksi määrittää objektin datatyypin (esimerkiksi kokonaisluku tai jono).
- Koodaus kuvaa missä muodossa tieto on.

(Simple Network Management Protocol 2011.)

2.11.2 Hallintatietokantataulut

SNMPv1 SMI määrittelee tarkasti jäsennellyt taulut, joita käytetään ryhmittämään taulukkomuotoinen objekti. Tällainen objekti sisältää monta muuttujaa. Taulut muodostuvat nollasta tai sitä useammasta rivistä, jotka on indeksoitu niin, että SNMP voi noutaa tai muuttaa kokonaisen rivin yhdellä Get, GetNext tai Set–komennolla. (Simple Network Management Protocol 2011.)

2.11.3 SNMP-versio 1 protokollatoiminnot

SNMP on periaatteeltaan yksinkertainen pyyntö/vaste-protokolla. Hallinta-asema lähettää pyynnön ja laite vastaa siihen. Tällainen toiminta ilmenee käytettäessä jotain seuraavista toiminnoista: **Get**, **GetNext**, **Set** ja **Trap**.

- **Get**-toiminto on hallinta-aseman keino hankkia näytteitä objektin agentilta. Jos Get-toimintoon vastaava agentti ei pysty toimittamaan kaikkia listassa olevia arvoja, se ei toimita mitään arvoja.
- **GetNext**-toiminto on hallinta-aseman käyttämä keino hankkia agentin taulussa tai listassa oleva seuraava objekti.
- Hallinta-asema käyttää **Set**-toimintoa määrittääkseen objektin sisältämän agentin arvot.
- **Trap**-toiminto on agenttien käyttämä epäsynkroninen tiedotuskeino hallinta-asemalle merkittävistä tapahtumista.
  (Simple Network Management Protocol 2011.)

2.12 SNMP-versio 2

SNMPv2 on v1:n kehittyneempi versio. Kuten SNMPv1 myös v2 toimii SMI:n sisällä. SNMPv2 tarjoaa parannuksia aiempaan versioon nähden mm. vaihtoehtoisia protokollatoimintoja. (Simple Network Management Protocol 2011.)

2.12.1 SNMP versio 2:n hallintatiedon (SMI) rakenne

SNMPv2 SMI (Structure of Managed Information) versio sisältää datatyyppien parannuksia joita ovat mm. bittijono, verkko-osoitteet ja laskurit. Kun SNMPv1 tukee vain 32-bittisiä verkko-osoitteita, niin SNMPv2 tukee myös muun tyyppisiä osoitteita. Myös 64-bittiset laskurit ovat määriteltyinä SNMPv2:ssa. (Simple Network Management Protocol 2011.)

2.12.2 SMI-tietoyksiköt

SNMPv2 SMI erittelee tietoyksiköt, jotka erittelevät toisiinsa liittyvien määritelmien ryhmän. On olemassa kolmentyyppisiä SMI-tietoyksiköitä: MIB-yksiköt (Management Information Base), mukautuvuuslausunnot ja kykenevyys lausunnot. MIB-yksiköt sisältävät keskinäisessä suhteessa olevat hallittavien objektien määritelmät. Mukautuvuuslausunnot tarjoavat järjestelmällisen tavan kuvailla hallittavien objektien ryhmää, jotka täytyy toteuttaa standardin mukaisesti. Kykenevyyslausuntoja käytetään ilmaisemaan agentin vaatiman tuen tarkkaa tasoa. Hallinta-

asema voi muuttaa suhtautumistaan agentteja kohtaan kykenevyyslausuntojen perusteella. (Simple Network Management Protocol 2011.)

2.12.3 SNMP versio 2 protokollatoiminnot

Get, GetNext ja Set-toiminnot, joita käytetään SNMPv1:ssä, ovat tarkalleen samat SNMPv2:ssa. SNMPv2 tarjoaa kuitenkin pieniä parannuksia protokollatoimintoihin. Esimerkiksi Trap-toiminto käyttää samaa funktiota SNMPv2:ssa, mutta viestit kulkevat eri muodossa. (Simple Network Management Protocol 2011.)

SNMPv2:ssa on myös kaksi uutta protokollatoimintoa: GetBulk ja Inform. GetBulk-toiminto on hallinta-aseman käyttämä tehokas isojen datablokkien hakutoiminto. Tällaisia isoja blokkeja voivat olla esimerkiksi moninkertaiset rivit taulussa. Inform-toiminto sallii yhden hallinta-aseman lähettää trap-tiedotteen toiselle hallinta-asemalle ja vastaanottaa tämän vastauksen. Jos agentti, joka vastaa GetBulk-toimintoon ei pystykään toimittamaan kaikkien muuttujien arvoja, se lähettää ne tiedot mitä sillä on. (Simple Network Management Protocol 2011.)

2.13 SNMP versio 1:n ja SNMP versio 2:n tietoturva

SNMP-versiot 1 ja 2 kärsivät heikosta tietoturvasta, mikä voi aiheuttaa yrityksille erilaisia tietoturvauhkia. Näitä uhkia ovat maskeeraukset, tiedon muokkaus, viestijonon ja ajoituksen muokkaus ja naamioituminen.

Maskeeraus tarkoittaa valtuuttamattoman tahon yritystä suorittaa hallintatoimintoja identiteettivarkauksilla. Tiedon muokkaus liittyy valtuuttamattoman tahon yrityksiin muuttaa valtuutetun tahon luomia viestejä. Viestijonon ja ajoituksen muokkausta ilmenee kun valtuuttamaton taho uudelleen tilaa, viivyttää tai kopioi ja myöhemmin toistaa valtuutetun tahon viestejä. Naamioitumisessa valtuuttamaton taho purkaa arvoja, jotka on varastoitu hallittaviin objekteihin tai se opettelee ilmoituspakon alaisia tapahtumia monitoroimalla hallinta-aseman ja agentin välisiä vaihtoja. (Simple Network Management Protocol 2011.)

2.14 SNMP versio 3

SNMPv3 on yhteensopiva standardipohjainen protokolla. Kahteen ensmmäiseen SNMP-versioon nähden, SNMPv3 tarjoaa selvästi parempaa tietoturvaa. Sen avulla on mahdollista tehdä suojattua yhteyksiä laitteisiin autentikoinnin ja salattujen pakettien avulla verkon yli. (SNMPv3 2011.)

SNMPv3:n turvallisuusominaisuuksia:

- Viestien eheys – Varmistetaan, että paketteihin ei ole tehty muutoksia siirron aikana.
- Autentikointi – Määritellään että viesti on oikeasta lähteestä.
- Salaus – Paketin sisällön sekoittaminen estää ulkopuolisia lähteitä näkemästä sitä. (SNMPv3 2011.)

SNMPv3 tarjoaa sekä turvallisuusmalleja, että turvatasoja. Turvallisuusmalli on autentikointi strategia, joka on tehty käyttäjälle ja ryhmälle jossa käyttäjä sijaitsee. Turvataso on sallittu turvallisuustaso turvallisuusmallin sisällä. Turvallisuusmallin ja turvatason yhdistelmä määrittelee, mikä turvallisuusmekanismi on käytössä, kun käsitellään SNMP-pakettia. (SNMPv3 2011.)

Havaintoja SNMPv3 objekteista:

- Jokainen käyttäjä kuuluu ryhmään.
- Ryhmä määrittelee kulkulupapolitiikan käyttäjäjoukolle.
- Kulkulupapolitiikka tarkoittaa mitä SNMP-objekteja voidaan käyttää lukemiseen, kirjoittamiseen ja luomiseen.
- Ryhmä määrittelee listan ilmoituksista, joita sen käyttäjät voivat vastaanottaa.
- Ryhmä määrittelee myös turvallisuusmallit ja turvatason käyttäjilleen. (SNMPv3 2011.)

SNMPv3 sisältää paljon muitakin parannuksia, kuin vain tietoturvallisuuteen liittyviä. Hallinnollisia ominaisuuksia aiempiin SNMP versioihin nähden ovat mm:

- SNMPv1 lähettää viestejä SNMP kokonaisuuksien välillä. Se tunnistaa nämä kokonaisuudet sovelluskokonaisuuksiksi ja protokollakokonaisuuksiksi. SNMPv3 vastaavasti nimeää uudelleen nämä sovellukset ja välineet.
- SNMPv1 ja SNMPv2 tarjoavat tunnistuksen perustuen yhteisöllisyyteen. SNMPv3 laajentaa tunnistuspalvelun sisältämään yksityisyyden.
- SNMPv1 käyttää kulunvalvontaa perustuen SNMP MIB näkymään. SNMPv3 määrittelee vastaavan konseptin nimeltään view-based access control model (VACM).
- VACM lisää kulunvalvonnan kaikkiin hallittaviin laitteisiin tarjoten myös hallintatietokannan objektitason hallintaa. Tämä antaa jokaisen SNMP-agentin valvoa kulkupolitiikkaa.
- Vaihtoehtoinen User-based Security Model (USM) tarjoaa käyttäjäkohtaisen tunnistuksen yksilöllisistä SNMP paketeista. Se tukee viestien tiivistelmiä var-

mistaakseen paketin eheyden. Se tarjoaa aikaleimavarmistuksen suojautuakseen toistuvia hyökkäyksiä vastaan. Se suojelee hallintatietoa salakuuntelulta käyttäen DES-salausta.

- SNMPv3 lisää yksilölliset nimet jokaiseen SNMP laitteeseen. Tämä toimenpide auttaa pakettien tunnistamisessa ja agenttien välisten suhteiden kartoittamisessa.
- SNMPv3 antaa hallinnoijien päivittää SNMP laitteiden asetukset automaattisesti käyttäen SNMP:tä. (Simoneau 1999, 242.)

2.15 Graafisten tilastojen työkalu

RRDToolin (Round Robin Database Tool) avulla voidaan päivittää lokitiedosto milloin vain halutaan. Lokitiedostolla tarkoitetaan tiedostoa, johon kerätään dataa monitoroiduista isäntälaitteista. RRDTool hakee automaattisesti tietolähteen arvot virallisissa aikajaksoissa ja kirjoittaa tämän datan lokitiedostoon. Uutta lokimerkintää tallennettaessa otetaan huomioon aiemmin tallennettu alkuperäinen data, joka mahdollistaa pitkäaikaisen seurannan. RRDToolin avulla on mahdollista kirjata dataa vaikka yhden minuutin välein vuoden ajanjaksolle, jolloin voidaan seurata kehitystä pidemmällä aikavälillä. Datan varastoiminen vuoden ajalta voi viedä melko paljon levytilaa ja aikaa tämän tiedon analysointiin. RRDTool tarjoaa ratkaisuksi mahdollisuutta määrittää millä aikavälillä datan keräystä suoritetaan ja mitä keräysfunktiota (keskiarvo, minimi, maximi, kokonaismäärä, viimeisin) käytetään. Myös datan kirjaamisen aikaväliä on mahdollista muuttaa. (RRDTool 2011.)

2.15.1 Data-arkisto

Datan arvot varastoidaan RRA-arkistoon (Round Robin Archives), mikä on erittäin tehokas tapa säilyttää dataa tietyn ajanjakson verran, kun tiedetään käytössä olevan levytilan olevan vakio. (RRDTool 2011.)

Kun halutaan varastoida esimerkiksi 1000 arvoa 5 minuutin välein, RRDTool varaa tilaa näille tuhannelle arvolle ja otsikkoalueelle. Otsikkoalueen tieto kertoo mitkä arvot kirjoitettiin viimeksi. Tässä esimerkkitapauksessa uusia arvoja kirjoitetaan vain tuhanteen arvoon saakka, koska niin alun perin haluttiin. Koska useita RRA-arkistoja voidaan määrittää yhteen RRD-tietokantaan, voit luoda toisen RRA:n varastoidaksesi esimerkiksi 750 data-arvoa 2 tunnin välein. RRA-arkistojen käyttö takaa sen, että tietokanta ei kasva ajan kuluessa, koska vanha data poistuu automaattisesti uuden tiedon tallentuessa sen tilalle. (RRDTool 2011.)

Jo aiemmin mainittujen keräysfunktioiden avulla voidaan varastoida juuri sen tyyppistä tietoa, mitä todella on tarpeellista tietää. Näitä ovat esimerkiksi prosessorin lämpötila, palvelimen kaatumisen ajankohta ja kaatumisen kokonaiskesto jne. (RRDTool 2011.)

2.15.2 Tunnistamaton tieto

RRD:n ominaisuuksiin kuuluu tallentaa tietoa tietyissä aikajaksoissa, kuten aiemmin mainittiin. Joskus on mahdollista, että tietoa ei ole saatavilla, kun on aika tallentaa sitä tietokantaan. Näissä tapauksissa tietokantaan tallennetaan tunnistamaton arvo. RRDToolin kaikki toiminnot tukevat tunnistamattoman arvon lisäystä. Luotaessa kiinteää tiedostoa, tunnistamattoman tiedon arvon määrä lasketaan ja kun uusi kiinteä arvo on valmis kirjoitettavaksi arkistoon, tarkistetaan että tunnistamattoman arvon prosenttiosuus on konfiguroitavan tason yläpuolella. Jos näin ei ole, tunnistamaton arvo kirjoitetaan arkistoon (RRA). (RRDTool 2011.)

2.15.3 Graafiset tilastot

RRDToolilla voidaan luoda numeerisia ja graafisia raportteja mistä tahansa RRD-tietokannasta, johon on tallennettu tietoa. Graafisia raportteja voidaan vapaasti muokata cactin kautta muun muassa koon, värin ja sisällön osalta. Graafiset tilastot ovat helppolukuisia ja paremmin ymmärrettäviä, kun verrataan kirjallisiin raportteihin. Graafisista tilastoista lisää luvussa 2.16.4.

2.16 Cacti

Cacti on PHP-koodikielellä rakennettu selaimella toimiva työkaluohjelmisto, joka tallentaa kaiken keräämänsä tiedon SNMP-protokollaa tukevista laitteista MySQL-tietokantaan. Näitä tietoja hyödyntäen Cacti osaa luoda graafisia tilastoja (graafit) sille määritellyistä toiminnoista, esimerkiksi verkkoliikenteen tai kiintolevyjen kapasiteetin valvonnasta. Graafisten tilastojen piirtäminen ja tallennus on mahdollista aiemmin mainitun RRDTool-työkalun avulla. Kuvissa olevat IP-osoitteet ja isäntänimet on viivattu yli mustilla palkeilla turvallisuussyistä. (What is Cacti? 2011.)

2.16.1 Tietolähteet

Cactin tietolähde-osioon (Data Source) voi määritellä jo olemassa olevien datankeräysscriptien lisäksi itse tehtyjä scriptejä. Näiden scriptien perusteella RRDTool kerää tietoja ajastetusti ja lisää ne MySQL-tietokantaan, josta se piirtää graafisia tilastoja, jotka cacti poimii omaan käyttöliittymäänsä nähtäväksi. (What is Cacti? 2011.)

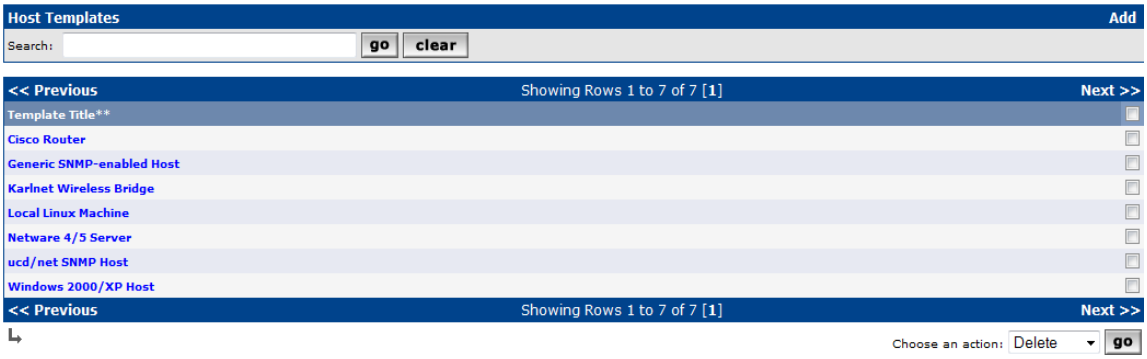
2.16.2 Datankeräys

- Datankeräystoiminto mahdollistaa käyttäjän omien scriptien käytön datankeräykseen. Jokainen scripti voi sisältää muuttujia, jotka täytyy syöttää jokaiselle scriptiä käyttävälle tietolähteelle erikseen, esimerkiksi IP-osoite.
- On myös mahdollista hankkia tietoa käyttäen SNMP:tä tai scriptiä hakemiston kanssa. Esimerkkinä voisi olla listan tekeminen IP-rajapinnoista tai asennetuista levyn osioista palvelimella.
- PHP-pohjainen laskin (poller) suorittaa scriptit, hankkii SNMP-datan ja päivittää RRD-tiedostot. (Features 2011.)

2.16.3 Pohjat

Mallipohjien avulla Cacti voi skaalautua suurelle määrälle tietolähteitä ja graafeja. Tämä sallii yksittäisen graafin tai tietolähteen mallipohjan luomisen, joka määrittää minkä tahansa graafin tai tietolähteen, joka käyttää sitä. Isäntäkohtaiset mallipohjat sallivat isännän toiminnallisuuksien määrittelyn. Näitä valmiita isäntäpohjia voidaan käyttää luodessa uusia isäntiä. Esimerkiksi kun lisätään Linux-isäntä, voidaan käyttää oletuspohjaa ”Local Linux Machine”, joka pitää sisällään mm. tietyt graafisten tilastojen pohjat. (Features 2011.)

Kuviossa 2 on lista Cactin valmiista isäntäpohjista. Lisättäessä uutta isäntälaitetta Cactiin voidaan valita käytettäväksi jokin seuraavista pohjista.



Kuvio 2.

2.16.4 Graafiset tilastot

Tietolähteistä, jotka vastaavat todellista dataa, voidaan RRDTool-työkalun avulla luoda rajaton määrä graafeja. Esimerkiksi käyttäjän halutessa monitoroida ping-viivettä, voidaan luoda valmista scriptiä käyttävä tietolähde, joka pingaa isäntälaitetta ja palauttaa arvon millisekunteina. Sen jälkeen, kun RRDToolille on määritelty asetukset kuinka dataa tulisi säilyttää, voi-

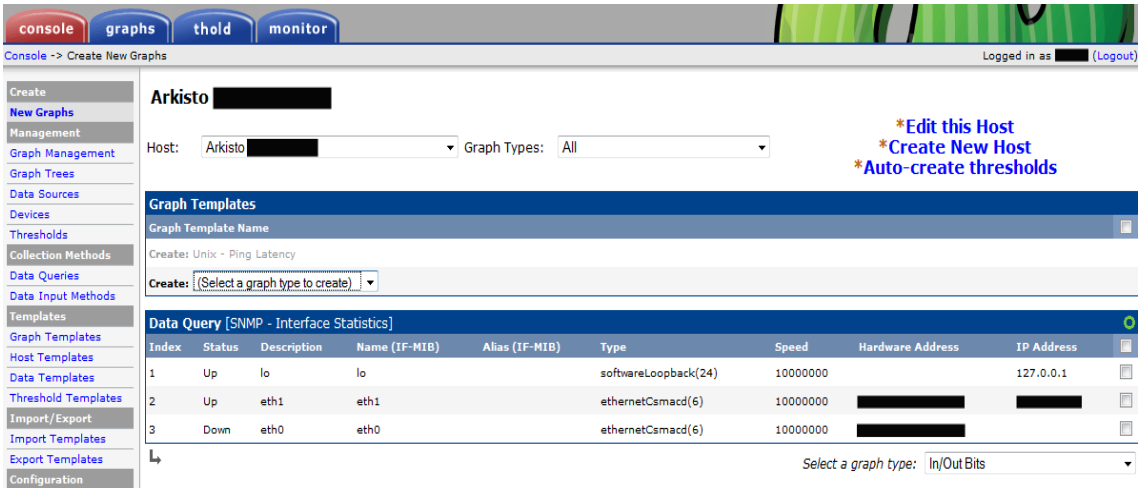
daan määrittää mitä tahansa lisätietoja, joita tietojen tulolähde edellyttää. Tässä tapauksessa kyseessä on määritelty isäntä ping-viiveen monitoroinnille. (What is Cacti? 2011.)

Perinteisen lista-näkymän lisäksi on mahdollista järjestää monitoroidut laitteet ja niistä piirretyt graafit hierarkkiseen puunäkymään, josta voidaan poimia haluttu isäntälaite. Tämä puunäkymä on erittäin käytännöllinen selkeytensä vuoksi.

Selainkäyttöliittymän kautta on mahdollista luoda uusia käyttäjiä ja hallita niitä. Käyttöoikeuksia voidaan määrittää erityyppisiä eri käyttäjille. Esimerkiksi jotkut käyttäjät voivat muokata graafisten tilastojen parametreja, kun taas toiset käyttäjät voivat vain katsella näitä graafeja. Kaikki käyttäjät voivat kuitenkin muokata graafien näkymät mieleisekseen.
(What is Cacti? 2011.)

Cacti sisältää valmiiksi suuren määrän valmiita graafisia pohjia, kuten kuvion 3 esimerkki kertoo. Valmista Pohjaa klikkaamalla pääsee muokkaamaan yksityiskohtaisesti graafin asetuksia.

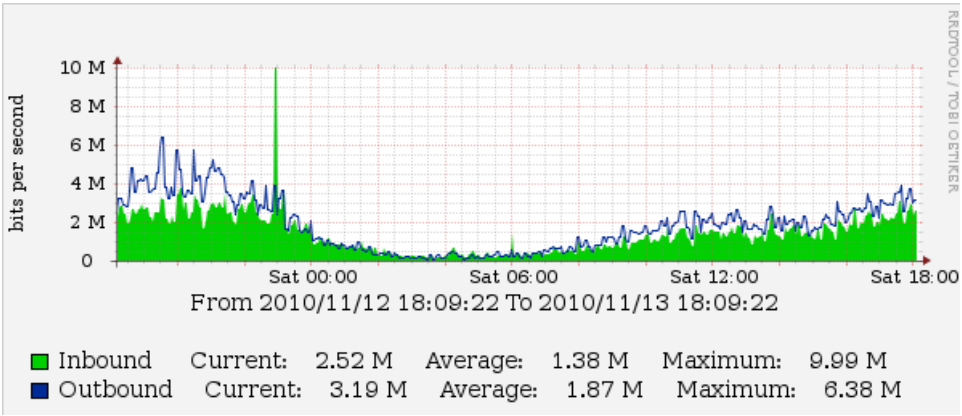
Kuvio 3.



Kuvio 4.

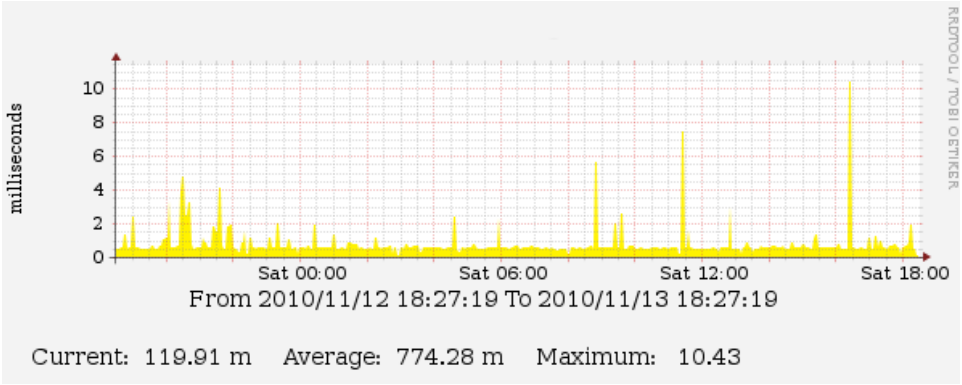
New Graphs-osiossa voi luoda uusia ja tarkastella isännille olemassa olevia graafeja, kuten kuvio 4 kertoo. Tässä näkymässä näkyvät myös graafit, jotka ovat down-tilassa, eli ne eivät ole käytössä sillä hetkellä. Isäntää voi vaihtaa ylhäällä olevasta pudotusvalikosta ja nähdä mitä graafeja millekin isännälle on todella olemassa.

Kuvioissa 5-8 on lähempi näkymä graafeista. Tilastot ovat selkeitä, helposti tulkittavissa ja värikoodien ansiosta ne on helppo erottaa toisistaan. Mitä tahansa graafia klikkaamalla pääsee näkymään, jossa voidaan tarkastella niiden päivittäistä, viikoittaista, kuukausittaista ja vuosittaista historiaa. Jokaisessa graafissa on zoomaustyökalu, jonka avulla voidaan tarkentaa aikajanaa hyvin tarkaksi; vaikka puolen minuutin tarkkuuteen.



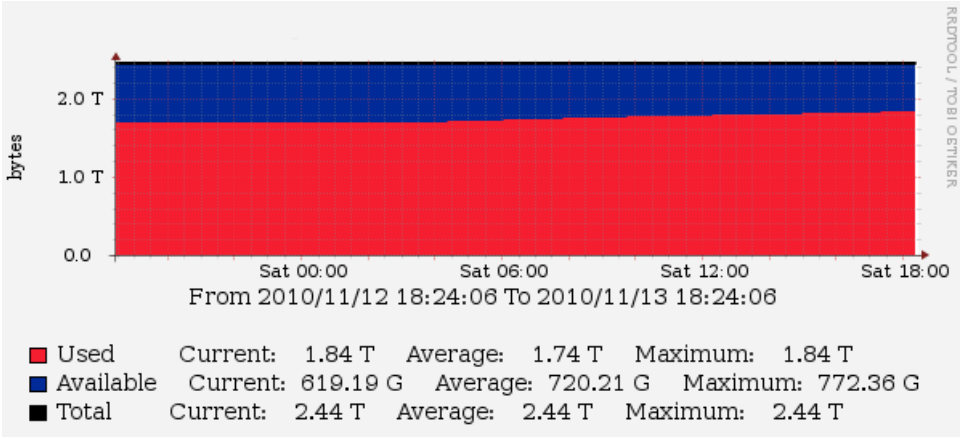
Kuvio 5. Verkkoliikenne.

Kuvio 5 kertoo sisään- ja ulospäin kulkevan verkkoliikenteen kaistan käytön bitteinä per sekunti. Graafista voidaan nähdä nykyinen, keskimääräinen ja maksimimaalinen kaistan liikenne.



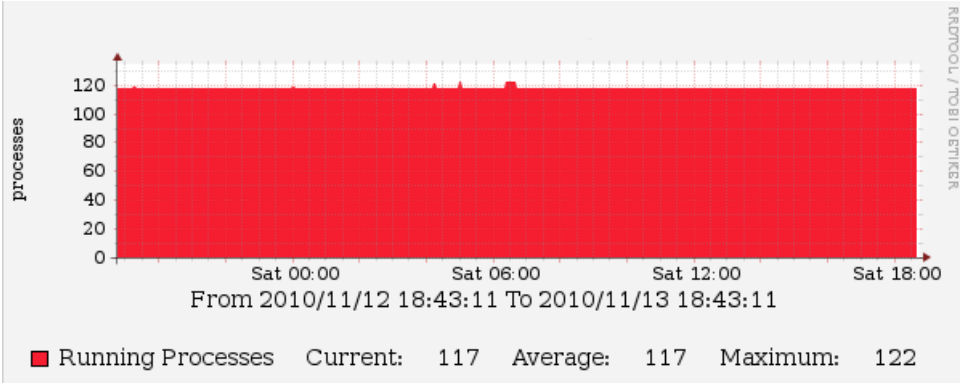
Kuvio 6. Ping-viive.

Kuvio 6 kertoo ping-viiveen palvelimelle millisekunteina.



Kuvio 7. Kiintolevytila.

Kuvio 7 kertoo kiintolevytilan kokonaismäärän, käytetyn määrän ja saatavilla olevan määrän. Yksiköinä käytetään giga- ja teratavuja.

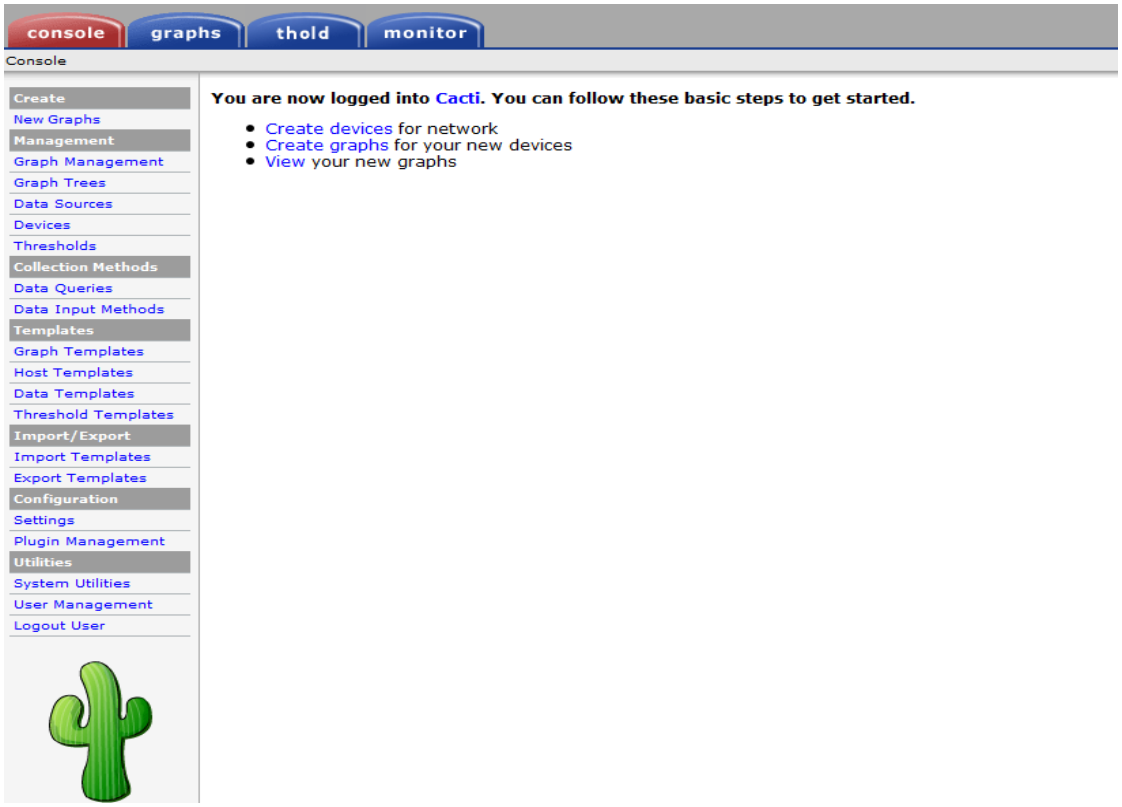


Kuvio 8. Prosessit.

Kuvio 8 kertoo käytössä olevien prosessien määrän ja myös keskiarvollisen ja maksimaalisen määrän.

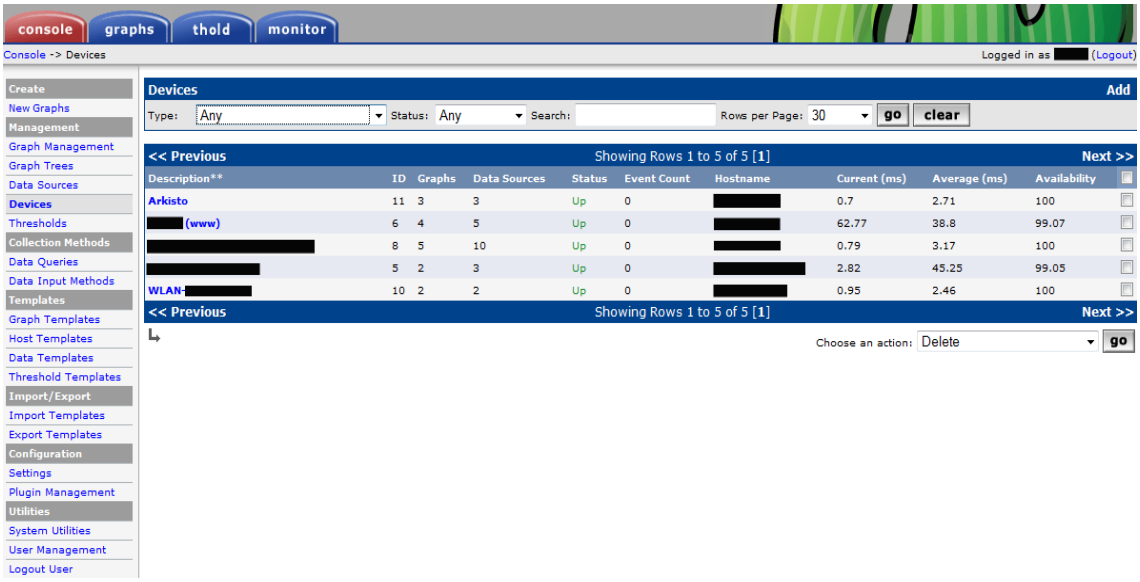
2.16.5 Cactin käyttöliittymä

Cactin käyttöliittymä koostuu perusrakenteeltaan sivun vasemmassa laidassa sijaitsevasta valikkopuusta, joka on nähtävillä niin kauan kuin pysytään konsolivälilehdellä. Päävalikot voivat olla aluksi termistönsä vuoksi hieman vaikeaselkoisia, mikä on kuitenkin vain tottumiskysymys.



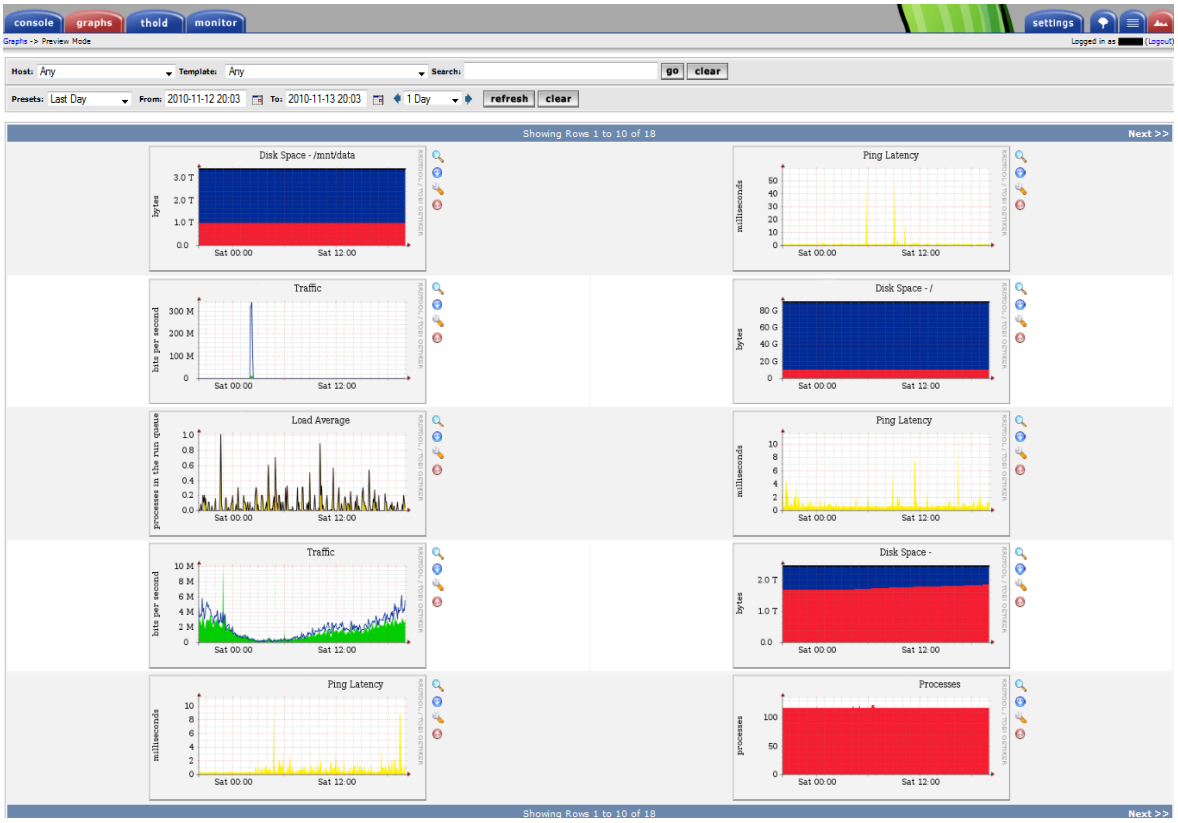
Kuvio 9. Cactin konsoli-ikkuna.

Kuviossa 9 on Cactin konsoli-ikkuna, eli perusnäkymä. Vasemmalla on valikkopuu, jonka avulla voidaan navigoida sivustolla. Sivun yläosassa ovat välilehdet, joilla pääsee kätevästi konsoli-ikkunaan, graafi-ikkunaan, thold -eli hälytysplugin-ikkunaan ja monitori-ikkunaan.



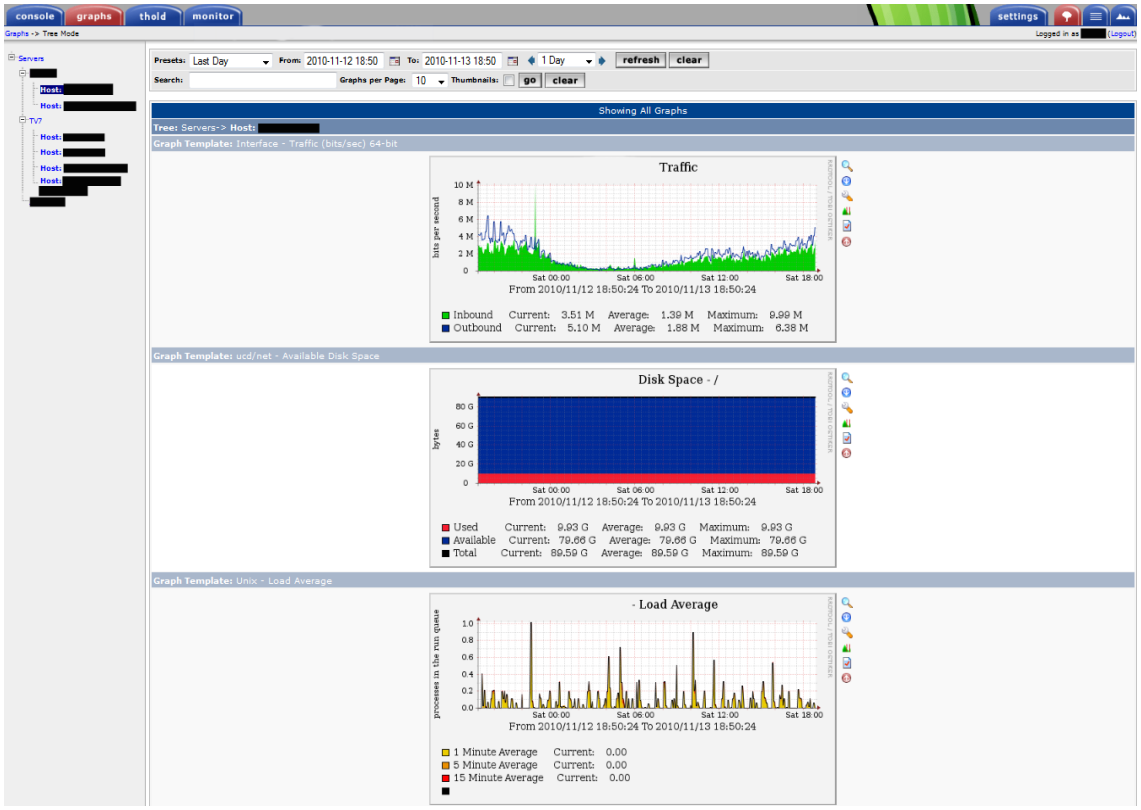
Kuvio 10.

Kuviossa 10 näkyy laitelista monitoroitavista isännistä. Isäntää klikkaamalla pääsee tarkempiin asetuksiin. Tässä osiossa tapahtuu uusien isäntälaitteiden luominen klikkaamalla add-painiketta.



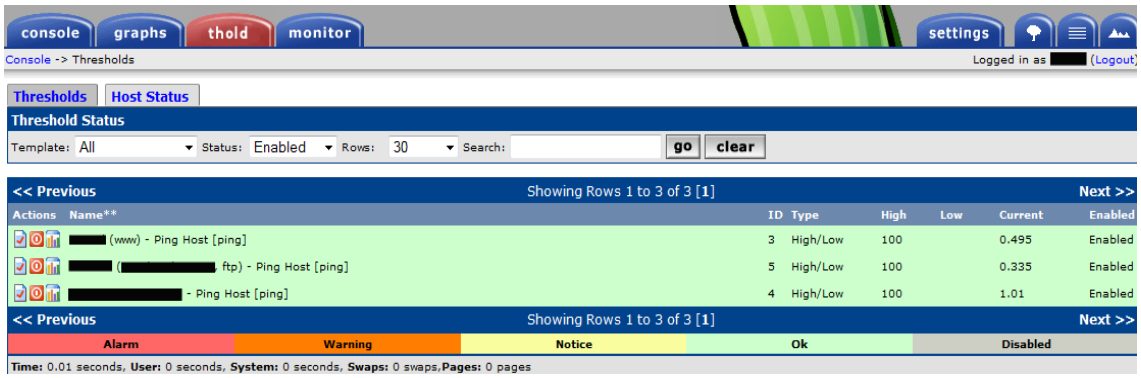
Kuvio 11.

Kuviossa 11 näkyy graafit-välilehti, jossa on valittuna preview view. Tässä näkymässä näkyvät kaikkien lisättyjen laitteiden graafit samassa ikkunassa. Tiettyjen laitteiden tiettyjä graafeja voidaan myös tarkastella tekemällä valinnat ylhäällä sijaitsevista pudotusvalikoista.



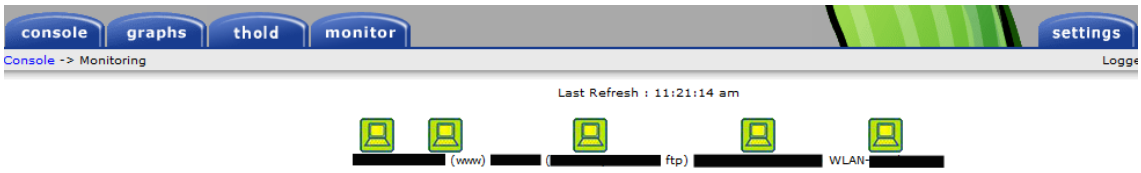
Kuvio 12.

Kuviossa 12 on ehkä selkein näkymä graafien katseluun. Näkymä on nimeltään tree view. Vasemmassa reunassa näkyy valikkopuu, josta valitaan haluttu isäntälaite, jonka jälkeen graafit listataan allekkain. Isäntälaitteita voidaan myös kätevästi jaotella eri kategorioihin. Laitteiden segmentointi helpottaa tietyn isännän löytymistä luettelosta, kun laitteita on paljon. Jaottelun avulla Cactiin voidaan lisätä hyvinkin suuri määrä monitoroitavia laitteita ilman, että käytettävyys kärsii. Tree view ja preview view-näkymän lisäksi on olemassa myös list view-näkymä, jossa graafit on listattu isäntineen allekkain ilman graafisten tilastojen esikatselua.



Kuvio 13.

Kuviossa 13 on hälytys-pluginin (Thold) näkymä. Listassa näkyvät isäntälaitteet, joihin on aktivoitu tässä tapauksessa ping-viiveen seuranta. Korkeimmaksi mahdolliseksi arvoksi on määritelty 100 millisekuntia. Jos tämä arvo ylittyy, Cacti lähettää siitä ilmoituksen sähköpostitse. Status-valikosta voi vaihtaa näkymää aktivoituihin, ei-aktivoituihin tai hälytystason saavuttaneisiin isäntiin. Current-osiosta näkee seurannan senhetkisen arvon.

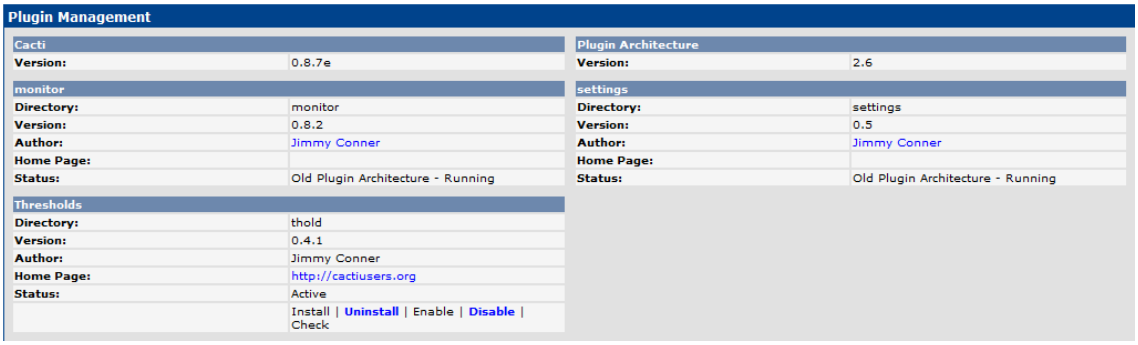


Kuvio 14.

Kuviossa 14 näkyvät kaikki valvonnassa olevat isäntälaitteet. Kyseessä on monitor-osio. Kun laitteet ovat vihreän värisiä, se tarkoittaa, että Cacti löytää nämä laitteet verkosta ja yhteydet toimivat. Jos laite on punaisena, yhteyksissä on jotain vialla. Tämä on nopea keino varmistaa palvelinten saatavuus.

2.16.6 Lisäosien hallinta

Cactiin on mahdollista ladata lisäosia eli niin sanottuja plugineja. Ennen lisäosien asentamista, täytyy Cactiin asentaa lisäosien arkkitehtuuri (Plugin Architecture), jonka avulla cacti pystyy tulkitsemaan pluginien toimintaa. Arkkitehtuurin asennus voi olla melko monimutkainen prosessi, mutta siihenkin on hyvin saatavilla dokumentaatioita, jotta selviää mahdollisista vastaantulevista ongelmista. Plugin-arkkitehtuurin asennus on välttämätön, jos halutaan saada automaattiset hälytykset toimimaan, sillä ainakaan Cactin versio 0.8.7e ei itsessään sisältänyt tällaista toiminnallisuutta. Tässä tapauksessa hälytyspluginina toimi thold-niminen lisäosa.



Kuvio 15.

Kuviossa 15 näkyy hälytyspluginin pelkistetyt asetukset, josta plugin voidaan asentaa/poistaa tai kytkeä päälle/pois. Näkymästä selviää myös lisäosan versionumero ja sen suunnittelija.

2.16.7 Hallintaohjelmiston valinta

Cacti valittiin projektiin hallintaohjelmistoksi, koska se on avoimen lähdekoodin ohjelmisto, joten siitä ei aiheutunut ylimääräisiä kuluja yritykselle. Cacti valittiin yrityksen esimiehen toimesta myös sen monipuolisuuden ja muokattavuuden takia. Kuten kappaleessa 2.16.1 on mainittukin, Cactin avulla voidaan ajaa myös omia scriptejä ja olemassa olevia voidaan muokata mieleisiksi. Itse kirjoitettujen scriptien etuna on se, että ne suorittavat juuri niitä toimintoja, joita itse halutaan niiden suorittavan. Lisäksi kattavan lisäosien tuen tarjoava Cacti-yhteisö oli vaikuttava tekijä ohjelmistoa valittaessa. Internetissä on ladattavissa lukuisa määrä erilaisia lisäosia, jotka voidaan asennuksen jälkeen ottaa suoraan käyttöön ilman suurempia muutoksia. Tämä säästää verkonhallinnoijan työaikaa, kun scriptejä ei tarvitse kirjoittaa itse. Myös se, että Cacti osaa käyttää RRDtool-työkalua graafisten tilastojen piirtämiseen, jotka selkeyttävät verkon valvontaa, on tärkeä ominaisuus ohjelmistossa.

3 Toteutus

Ennen projektin alkua SNMP-hallintaohjelmisto oli jo valittu, joten oikean ohjelmiston etsimiseen ei tarvinnut uhrata aikaa. Hanke aloitettiin asentamalla Cacti-ohjelmisto ja muut vaadittavat ohjelmalliset komponentit yhteen Linux-pohjaisista tuotantopalvelimista. Cactin ja muiden komponenttien asennus sujui varsin nopeasti ja ilman ongelmia. Kaikkien tarvittavien asennusten jälkeen päästiin tutustumaan itse ohjelmistoon ja sen toimintoihin tarkemmin. Aluksi isäntälaitelistaan lisättiin testikäyttöä varten perusmallin työpöytäkone, joka asetettiin automaattiseen ympärivuorokautiseen valvontaan.

Cactin testaaminen tapahtui kokeellisella ajanjaksolla, jonka aikana sen toimintaa seurattiin automaattisen valvonnan lisäksi manuaalisesti lähes päivittäin. Työasemaa testattiin ja moni-

toroitiin noin kahden kuukauden ajan. Seurannan aikana Cacti toimi sujuvasti, se piirsi sille määritellyt graafiset tilastot ongelmitta ja niitä päästiin kätevästi tarkastelemaan graafi-osion kautta.

Kun todettiin, että RRDTool-työkalun tietokantaan tallentama tieto oli paikkansapitävää esimerkiksi ping-viiveen ja verkkokaistan liikenteen osalta, alettiin miettiä automaattisten hälytysten aktivointia. Cacti itsessään ei sisältänyt hälytyksiä mahdollistavaa funktiota, joten kyseinen toiminnallisuus asennettiin lisäosana. Lisäosien käyttöä varten asennettiin ensiksi lisäosien arkkitehtuuri ja tämän jälkeen voitiin asentaa hälytykset mahdollistava lisäosa. Onnistuneen asennuksen jälkeen hälytysarvot säädettiin seurannan asetuksiin siten, että hälytys oli väistämätön. Tällä tavoin saatiin selville, että Cacti varmasti reagoi esimerkiksi hälytysarvojen ylittyessä ja lähettää ilmoituksen kaikille määritellyille sähköpostiosoitteille. Monitoroitavan laitteen kaatumistilakin tuli testattua muutaman kerran viikossa tietokoneen sammumisen yhteydessä. Kun tietokone sammui, samalla katkesi myös yhteys Cactiin ja parin minuutin kuluttua isäntälaitteen kaatumisesta tuli sähköpostiin hälytys.

Kun hallintajärjestelmän todettiin monitoroivan työasemaa onnistuneesti, lisättiin Cactin isäntälaitelistaan yksi tuotantopalvelimista ja tehtiin sille samat perustestaukset kuin työasemakoneellekin lukuun ottamatta palvelimen kaatumisilmoituksen testausta. Cacti määriteltiin piirtämään vastaavat perustoimintojen graafiset tilastot palvelimelta, kuin työasemakoneella oli määritelty ja alettiin seurata niiden toimivuutta. Kun datankeräyksen ja graafisten tilastojen luomisen todettiin olevan toimivaa, lisättiin tuotantopalvelimeen enemmän graafisia tilastoja piirrettäväksi, jotta saatiin testattua niiden kaikkien todenmukainen toiminta. Prosessorin kuormituksen seuranta ei suoraan toiminut oikein millään monitoroitavalla koneella. Kuormituksen prosentuaalinen määrä näytti olevan eriluokkaa kuin tietokoneen käyttöjärjestelmän kautta tarkasteltuna. Tämä osio on edelleen tarkemman seurannan alla, eikä siihen voi vielä täysin luottaa.

Kiintolevyjen datan keräys toteutettiin aluksi samba-protokollan kautta. Tarkemman pohdinnan jälkeen datan keräys suoritettiin kuitenkin Linux-palvelimen oman mount-pisteen kautta. Ei ole syytä kuljettaa tietoa sekä SNMP- että samba-protokollan lävitse, jos tieto voidaan hankkia pelkästään SNMP-protokollan avulla. Samba-protokollan kaatuessa tulisi vain ylimääräisiä katkoksia datan keräykseen.

Verkkoliikenteen seurannassa havaittiin aluksi ristiriitaisuuksia kerättyjen arvojen suhteen. Käytössä oli 32-bittinen bittejä/sekunti arvoilla toimiva seuranta. Kun monitorointi muutettiin 64-bittiseksi, alkoi RRDTool piirtää tietokannasta todenmukaista tietoa graafisiin tilastoihin. Tarkkaa syytä tähän ei löydetty. Ilmeisesti käyttöjärjestelmän 64-bittisyys vaikutti asiaan jollain tavalla.

Kun yhden tuotantopalvelimen automaattisen monitoroinnin havaittiin olevan graafien ja automaattisten hälytysten osalta kunnossa, konfiguroitiin valvonnan piiriin kaksi tuotantopalvelinta lisää.

3.1 Cactin ja tarvittavien komponenttien asennus

Seuraavat ohjelma-komponentit asennettiin palvelimelle, joka toimi SNMP-järjestelmän hallinta-asemana. Kyseessä oli Linux-palvelin ja kaikki asennukset saatiin hoidettua kätevästi komentorivin kautta.

Apachen asennus: apt-get install apache2

PHP-osien asennus: apt-get install php5
apt-get install php5-mysql
apt-get install php5-snmp

MySQL-tietokannan asennus: apt-get install mysql-server

SNMP-protokollan asennus: apt-get install snmp

Cactin asennus: apt-get install cacti

Yllä olevat komennot ovat asennuskomentoja. Raportin liite-osiosta löytyvät Cacti-manuaalista poimitut täydelliset ohjeistukset asetusten määrittämiseksi. Myös plugin-arkkitehtuurin asennusohjeet löytyvät liitteenä. Kaikki asetus- ja asennusohjeet ovat englanninkielisiä.

3.2 Laitteen lisääminen

Seuraavassa on esimerkki laitteen lisäämisestä Cactin isäntälaitelistaan.

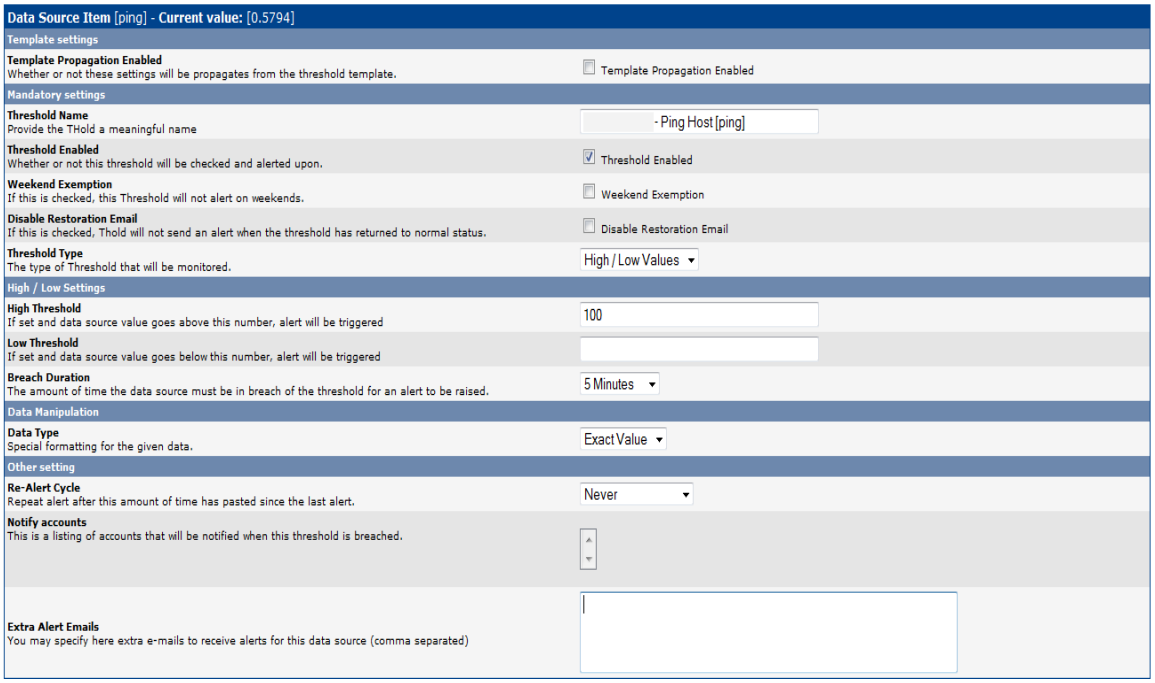
1. Määritellään isäntälaitteelle sitä parhaiten kuvaava nimi.
2. Kerrotaan Cactille isäntälaitteen dns-nimi tai IP-osoite.
3. Voidaan valita isännälle oletuspohja joka määrittelee, minkälaista tietoa laitteesta tullaan keräämään ja mitä graafeja niistä piirretään. Tässä tapauksessa valittiin local linux machine. Oletuspohjaa ei ole pakollista valita jos haluaa itse tarkasti määritellä halutut tietolähteet ja graafit. Oletuspohjiin voi myös lisätä jälkeenpäin haluttuja tietolähteitä ja niiden graafeja tai poistaa niitä.

4. Isäntä voidaan myös kytkeä pois päältä rastittamalla Disable Host jos niin halutaan. Jos isäntää aiotaan monitoroida, kuten tässä tapauksessa haluttiin, tulee rastittaa kohta Monitor Host.
5. Cactille pitää kertoa tapa, jolla tarkastetaan isäntälaitteen saatavuus. Vaihtoehtoina ovat SNMP-protokolla, ping-pyyntö, SNMP ja ping ja SNMP tai ping. Tässä tapauksessa haluttiin valita SNMP ja ping, jotta saatavuuden tarkastelu olisi mahdollisimman toimivaa. Muiden saatavuuteen liittyvien asetusten suhteen käytettiin oletusasetuksia.
6. Seuraavaksi tulee valita SNMP-protokollan versio, jota käytetään. Tässä työssä käytettiin SNMPv2:ta. Version valinnan jälkeen luodaan yhteisönimi, joka toimii myös ikään kuin salasanana, jos se määritellään yksilölliseksi. Oletusyhteisöä voidaan myös käyttää. Klikataan Create-painiketta ja uusi isäntä on luotu laitelistaan.
7. Klikkaamalla isäntälaitetta päästään muuttamaan yllämainittuja asetuksia milloin vain on tarpeen.

3.3 Hälytykset

Jos monitoroitavilla palvelimilla tapahtuu jotain tavallisuudesta poikkeavaa toimintaa, voidaan Hälytyslisäosan avulla saada hälytysviestejä sähköpostiin. Hälytysten asettaminen vaatii oman erillisen pluginin, joka pitää asentaa Cactin perusasennuksen jälkeen. Jotta olisi mahdollista asentaa plugineja, tulee sitä ennen asentaa plugin-arkkitehtuuri samalle hallintaasemalle, jossa Cacti sijaitsee. Kun Lisäosien arkkitehtuuri ja haluttu lisäosa on asennettu, voidaan konfiguroida asetukset Cactin kautta hälytysten aktivoimiseksi.

Tässä tapauksessa hälytys-pluginina toimii Thold-niminen plugin. Kuviossa 16 on avattu pingviiveen asetuksien määrittely hälytyksiä varten. Kenttiin syötetään halutut arvot, jotka tässä tapauksessa ylittyessään laukaisevat hälytyksen ja tieto siitä tulee sähköpostiosoitekenttään määriteltyihin osoitteisiin. Extra Alert Emails- kenttään voidaan syöttää useampi sähköpostiosoite, johon halutaan Cactin lähettävän hälytysviestit.



Kuvio 16.

4 Yhteenveto

Tässä yhteenvedossa on käytetty yrityksen esimiehen lausuntoja projektista. Loppupäätelmät pohjautuvat pienimuotoiseen haastatteluun, joka tehtiin vuoden 2011 alussa hieman ennen tämän raportin valmistumista. Haastateltavana toimi projektin esimies Ojares.

Projektin päätavoitteena oli mahdollistaa yrityksen käytössä olevien palvelinten keskitetty seuranta yhden hallintajärjestelmän kautta. Palvelinten määrän takia on ollut työlästä seurata niitä jokaista erikseen. Uusi hallintajärjestelmä on tehnyt palvelinten manuaalisen seurannan erittäin selkeäksi ja vaivattomaksi, koska käytössä on vain yksi järjestelmä, johon kirjaudutaan kertaalleen ja sen jälkeen voidaan tarkastella jokaisen palvelimen meneillään olevaa toimintaa ja toimintahistoriaa vain siirtymällä valikkopuussa halutun isäntälaitteen tietoihin. Informaation suuresta määrästä johtuen palvelinten käyttäytymisen seuranta on vienyt aiemmin paljon aikaa. Cactin käyttöönoton myötä informaation hahmottaminen on helpottunut, koska kaikki tieto on hyvin esillä, helposti saatavissa ja selkeästi integroitu yhden järjestelmän sisälle. Pääsy käsiksi tietoon on helpompaa ja tiedon käsittely on nopeampaa.

Uudessa järjestelmässä tiedon määrän hahmottamisessa ja tulkitsemisessa ovat apuna graafiset tilastot. Aiemmin palvelinten toimintaa tarkasteltiin muun muassa lokitiedostoihin tallennetun tiedon perusteella. Lokien lukeminen ja tulkitseminen voi olla työlästä ja vaikeaselkoista, koska tieto esitetään tekstipohjaisesti ja useasti rivitetään allekkain ilman välilyöntejä. Cactin ja RRDTool-työkalun avulla saatiin käyttöön helposti tulkittavat ja hyvin havainnolliset

graafiset tilastot. Jotta graafien erottaminen toisistaan olisi mahdollisimman selkeää, ne esitetään oletuksellisesti kukin tietyllä värikoodilla, jonka voi muuttaa mieleisekseen. Graafiset tilastot tekevät palvelinten tarkastelusta selkeää ja mielekästä työtä. Lokitiedostojen tarkastelu ei kuitenkaan ole jäänyt kokonaan historiaan, mutta nyt käytössä on myös vaihtoehtoinen ratkaisu.

SNMP-järjestelmän käyttöönotto on mahdollistanut palvelinten automaattisen monitoroinnin. Projektin tavoitteena automaatiotekniikan mahdollistamana oli aktivoida automaattiset hälytysviestit. Hälytysviestejä varten määriteltiin Cactiin tiettyjen komponenttien ja toimintojen hälytysarvot, joiden ylittyessä saatiin tieto sähköpostiin parin minuutin viiveellä. Hälytykset toimivat hyvin, eikä niiden suhteen ollut ongelmia. Hälytysviesti palvelimen kaatumisestakin saatiin testattua käytännössä, kun yksi palvelimista piti ajaa alas epävakaan toiminnan seurauksena. Tieto palvelimen kaatumisesta tuli nopeasti sähköpostiin, mikä antoi luottamusta järjestelmän toimivuudesta. Testaamalla tehtyjen vertailujen jälkeen SNMP-protokollan avulla kerätty tieto osoittautui luotettavaksi. Testauskohteina olivat muun muassa up/down –tila, ping-viive, prosessien lukumäärä ja kiintolevytila.

Automaattisen valvonnan hyötyjä yritykselle ovat ajalliset säästöt, koska voidaan luottaa järjestelmän kykyyn valvoa palvelimia itsenäisesti ja tilanteiden sattuessa saataisiin hälytysviestit. Kun kaikki toimii niin kuin pitääkin, Cacti ei lähetä viestejä ja palvelinten tarkkailu verkon ylläpitäjän osalta on vähäistä, mikä taas säästää aikaa muihin työtehtäviin. Testauksien perusteella automaattinen valvonta nopeuttaa reagointia ja pienentää viiveitä ongelmien korjaamisessa. Viat voidaan paikallistaa nopeammin. Voidaan siis puhua todellisista saavutuksista kustannussäästöjen suhteen, koska työntekijöiden toimesta valvontaan kulutetaan vähemmän aikaa ja vikojen korjaamiseen kuluu vähemmän työtunteja.

Uusi järjestelmä on hyvä työkalu ennaltaehkäisevään toimintaan. Jos jokin palvelimista tai niiden komponenteista uhkaa hajota, siitä todennäköisesti näkyy merkkejä Cactin seurannassa. Tällä tavoin voidaan reagoida ennakoivasti tilanteeseen ja korvata mahdollisesti vioittuneet komponentit ennen kuin ne ovat käyttökelvottomia.

Projektista oli tarkoitus selviytyä ilman ylimääräisiä kustannuksia. Tähän tavoitteeseen päästiin, koska käyttöön otettiin avoimen lähdekoodin ohjelmisto Cacti, joka on täysin ilmainen ohjelmisto. Cacti asennettiin yhteen olemassa olevista tuotantopalvelimista, joten kustannuksia ei kertynyt laitteistonkaan suhteen. Hanke ei ole nähtävästi tuonut mukanaan minkäänlaisia haittoja, jotka vaikuttaisivat palvelinten tai verkon toimintoihin. Ilmaiseksi ohjelmistoksi Cactin ominaisuudet ja toimivuus ovat hyvällä tasolla. Maksullisista verkonhallintasovelluksista yrityksellä ei ole kokemusta, mutta tämän tasoisen järjestelmän kustannussäästöjen luulisi kiinnostavan maksullisten ohjelmistojen käyttäjiäkin.

Järjestelmän täysi käyttöönotto odottaa vielä sopivaa hetkeä. Monitoroinnin piirissä on nyt muutama palvelin, jotka ovat ympärivuorokautisessa valvonnassa ja niiden suhteen Cactin toiminta on ollut toistaiseksi kiitettävällä tasolla. Lähiaikoina on tarkoitus lisätä loputkin palvelimet monitoroitavaksi ja hankkia lisää tarpeellisia tiedonkeräyslähteitä kuten komponenttien lämpötilojen seuranta ja asettaa prosessorikuorman seuranta toimivaksi.

Täysi siirtyminen järjestelmän käyttöön vaatii hieman organisointia. Manuaalinen tarkkailu tulee olemaan edelleen tärkeässä roolissa verkonhallinnassa, joten ylläpidollisen prosessin rakentamisessa tulee miettiä muun muassa miten valvontaa jaetaan it-henkilöstön kesken. Kuka tarkkailee? Milloin tarkkailee? Mihin osa-alueisiin tarkkailussa tulisi keskittyä?

Ennakoivan toiminnan suhteen Cactin käyttöönotto on kasvattanut yleisellä tasolla tietoisuutta palvelinten ja verkon nykytilasta. Pidemmällä aikavälillä tietoisuus kasvaa edelleen yhä yksityiskohtaisempaan suuntaan ja voidaan jo alkaa hyödyntämään järjestelmää verkon laajentamisen suunnittelussa.

Tavoitteena on ottaa täysi hyöty irti uudesta valvontajärjestelmästä ja lisätä valvonnan piiriin palvelinten lisäksi kaikki yrityksen SNMP-protokollaa tukevat laitteet ja henkilöstön työpöytäkoneet. Näitä laitteita on paljon, mutta tällä tavoin saataisiin selkeämpi kuva koko yrityksen verkon rakenteesta. Kun Cactiin lisätään kaikki verkon laitteet, sitä voisi käyttää eräänlaisena lisämanuaalina verkon rakenteen määrittämisessä. Näille laitteille voitaisiin antaa niitä parhaiten kuvaavat nimet esimerkiksi valmistajan, tarkan mallin ja muiden oleellisten tietojen mukaan.

Israelin toimipisteessäkin olevat palvelimet ja laitteisto olisi hyvä laittaa automaattiseen valvontaan, koska ohjelmatuotanto on siellä kysynnän suhteen melko suurta luokkaa ja tärkeä osa ohjelmatuotannon kokonaisuutta.

Alustavien tutkimusten perusteella hälytysviestit on myös mahdollista saada esimerkiksi älypuhelimiin melko vaivattomasti tekstiviestin muodossa. Yksi mahdollisuus on hankkia puhelimeen tarkoitukseen sopiva ohjelmisto, mikä löytyi ainakin Applen OS X-mobiilikäyttöjärjestelmälle. Noin neljän euron hintainen PushMail-niminen ohjelmisto välittää automaattisesti halutut sähköpostiviestit tekstiviestin muodossa puhelimeen. Tällä tavoin saataisiin ongelmatilanteissa tieto asianomaisille hyvin nopeasti. Tarkoituksena on testata erilaisia vaihtoehtoja tekstiviestihälytysten aikaansaamiseksi. Edellä mainittu ohjelmisto vaikuttaa erittäin vaivattomalta ja toimivalta ratkaisulta. Sen toimivuus tullaan testaamaan hyvin pian ja jos siihen ollaan tyytyväisiä, ei välttämättä ole tarvetta etsiä muita ratkaisuja.

Lähteet

Features. Viitattu 7.1.2011.
http://www.cacti.net/features.php

Hakala, M. & Vainio M. 2002. Tietoverkon rakentaminen. Porvoo: WS Bookwell.

Installing the Plugin Architecture. Viitattu 2.5.2011.
http://docs.cacti.net/manual:087:1_installation.9_pia

McCarty, B. 2005. Linux – Fedora & Red Hat Enterprise Linux – Tehokas hallinta. Jyväskylä: Gummerus Kirjapaino Oy.

Ojares, J. 2011. Yhteenveto-osion haastattelu. 25.4.2011.

Planet KVM. Microdata. Viitattu 28.12.2010.
http://www.microdata.fi/pdf/Microdata/Microdata_KVM.pdf

Puska, M. 2001. Linux palvelimena. Pieksämäki: Talentum

RRDTool. Viitattu 13.03.2011.
http://oss.oetiker.ch/rrdtool/doc/rrdtool.en.html

Simoneau, P. 1999. SNMP network management. New York: McGraw-Hill.

Simple Network Management Protocol. Viitattu 22.01.2011.
http://docwiki.cisco.com/wiki/Simple_Network_Management_Protocol

SNMPv3. Viitattu 10.2.2011.
http://www.cisco.com/en/US/docs/ios/12_0t/12_0t3/feature/guide/Snmp3.html

SSH Port Forwarding. Viitattu 15.12.2010.
http://www.symantec.com/connect/articles/ssh-port-forwarding

The Cacti Manual. Viitattu 2.5.2011
http://www.cacti.net/downloads/docs/pdf/manual.pdf

What is Cacti? Viitattu 19.1.2011.
http://www.cacti.net/what_is_cacti.php

Liitteet

Liite 1: Cactin ja lisäosien arkkitehtuurin asennusohjeet

Seuraava manuaali on suora lainaus cacti manuaalista (The Cacti Manual).

Installing Under Unix

Please make sure, the following packages are installed according to your operating systems requirements. Verify, that httpd and mysqld are started at system startup.

Required Packages for RPM-based Operating Systems

- httpd
- php
- php-mysql
- php-snmp
- mysql
- mysql-server
- net-snmp

Ports for FreeBSD

- www/apache2
- net/rrdtool
- net/net-snmp
- www/php4-cgi
- lang/php4 (With MySQL and SNMP Support)
- databases/mysql323-server

Configure PHP

Please ensure, that PHP support is either builtin or installed for the following PHP extension modules:

- mysql (For configuration, see note below)
- SNMP (For configuration, see note below)
- XML
- Session
- Sockets
- LDAP (Required only when using LDAP authentication)
- GD (Required only for some Plugins)

You may run the following command to get the list of all available PHP modules
php –m

We will continue using the most recommended way of configuring php extension modules. Find the file /etc/php.ini and make the following changes to it:

extension_dir = /etc/php.d

This will enable PHP to find more configuration directives in that very directory. Other distros point to /usr/lib/php/modules instead. In each case, you should locate e.g. mysql.so in that directory.
Activate the MySQL extension via /etc/php.d/mysql.ini

; Enable mysql extension module
extension=mysql.so

Activate the SNMP extension via /etc/php.d/snmp.ini

; Enable snmp extension module
extension=snmp.so

If using PHP 4.3.5 or less include the following line. If using 4.3.6 or greater, you should remove this line if present.

session.save_path=/tmp

If you want to allow template importing, uncomment the following line:

file_uploads = On

Configure the Webserver (Apache)

If you are using Apache 1.3.x, installation of PHP 5 is not recommended.
Please find the file /etc/httpd/conf/httpd.conf or equivalent and make the following changes to it:

Load config files from the config directory "/etc/httpd/conf.d".
Include conf.d/*.conf

Now, please locate the PHP configuration file at /etc/httpd/conf.d/php.conf
If using PHP 5, then add the following lines.

```
# PHP is an HTML-embedded scripting language which attempts to make it
# easy for developers to write dynamically generated webpages.
LoadModule php5_module modules/libphp5.so
#
# Cause the PHP interpreter to handle files with a .php extension.
AddHandler php5-script .php
AddType text/html .php
#
# Add index.php to the list of files that will be served as directory
# indexes.
DirectoryIndex index.php
```

Configure MySQL

Set a password for the root user

```
shell> mysqladmin --user=root password somepassword
shell> mysqladmin --user=root --password reload
```

Install and Configure Cacti

1. Extract the distribution tarball.

```
shell> tar xzvf cacti-version.tar.gz
```

2. Create the MySQL database:

```
shell> mysqladmin --user=root create cacti
```

3. Import the default cacti database:

```
shell> mysql cacti < cacti.sql
```

4. Optional: Create a MySQL username and password for Cacti.

```
shell> mysql --user=root mysql
mysql> GRANT ALL ON cacti.* TO cactiuser@localhost IDENTIFIED BY 'somepassword';
mysql> flush privileges;
```

5. Edit include/config.php and specify the database type, name, host, user and password for your Cacti configuration.

$database_type = "mysql";
3Chapter 2. Installing Under Unix
$database_default = "cacti";
$database_hostname = "localhost";
$database_username = "cactiuser";
$database_password = "cacti";

6. Set the appropriate permissions on cacti's directories for graph/log generation. You should execute these commands from inside cacti's directory to change the permissions.
shell> chown -R cactiuser rra/ log/
(Enter a valid username for cactiuser, this user will also be used in the next step for data gathering.)

7. Add a line to your /etc/crontab file similar to:

*/5 * * * * cactiuser php /var/www/html/cacti/poller.php > /dev/null 2>&1
Replace cactiuser with the valid user specified in the previous step.
Replace /var/www/html/cacti/ with your full Cacti path.

8. Point your web browser to:
http://your-server/cacti/
Log in with a username/password of admin. You will be required to change this password immediately. Make sure to fill in all of the path variables carefully and correctly on the following screen.

(Optional) Install and Configure Spine

Spine is a very fast poller engine, written in C. It is an optional replacement for cmd.php. If you decide to use it, you will have to install it explicitely. It does not come with cacti itself. The easiest way is to install Spine using rpm or ports. You will find packages for Spine at the main cacti site or from your distribution.
To compile Spine, please download it to any location of your liking. Then, please issue from the downloaded directory following commands

shell>aclocal
shell>libtoolize --force (glibtoolize --force on Max OS)

shell>autoheader
shell>autoconf
shell>automake
shell>./configure
shell>make
shell>make install

Assuming, you've managed to install Spine correctly, you will have to configure it. The configuration file may be placed in the same directory as Spine itself or at /etc/Spine.conf.

DB_Host 127.0.0.1 or hostname (not localhost)
DB_Database cacti
DB_User cactiuser
DB_Password cacti
DB_Port 3306

All other pre 0.8.6 settings are obsolete.

Apply Patches

Please visit the Cacti website at http://www.cacti.net/download_patches.php If any patch has been released, you will find installation instructions there.
As an example, please find patch installation instructions for cacti 0.8.6j here. Do not apply those patches to recent releases!

wget
http://www.cacti.net/downloads/patches/0.8.6j/ping_php_version4_snmpgetnext.patch
wget http://www.cacti.net/downloads/patches/0.8.6j/tree_console_missing_hosts.patch
wget http://www.cacti.net/downloads/patches/0.8.6j/thumbnail_graphs_not_working.patch
wget http://www.cacti.net/downloads/patches/0.8.6j/graph_debug_lockup_fix.patch
wget http://www.cacti.net/downloads/patches/0.8.6j/snmpwalk_fix.patch
patch -p1 -N < ping_php_version4_snmpgetnext.patch
patch -p1 -N < tree_console_missing_hosts.patch
patch -p1 -N < thumbnail_graphs_not_working.patch
patch -p1 -N < graph_debug_lockup_fix.patch
patch -p1 -N < snmpwalk_fix.patch

You might need to reapply file/folder security on the files patched. Double check they are correct. Please pay attention not to break cacti when using SELinux or using NTFS file security. If you encounter

PHP Warning: include_once(./lib/html_tree.php) [function.include-once]:
failed to open stream: Permission denied in /var/www/cacti/graphs.php on line 33, referer: http://localhost/cacti/graphs.php
or the like, it is very likely that your permissions are wrong.

Seuraavana on suora lainaus cactin lisäosien arkkitehtuurin dokumentaatiosta. (Installing the Plugin Architecture)

Installing the Plugin Architecture (PIA)
The Plugin Architecture (PIA) is a set of code changes to core cacti. It is provided by Jimmy Conner (cigamit), one of the Cacti core developers. The Plugin Architecture for Cacti was designed to be both simple in nature and robust enough to allow freedom to do almost anything in Cacti. Cacti itself is designed nicely enough that integrating into it is fairly easy with very little modifications necessary. Eventually Cacti will come with a standard plugin architecture that will allow you to create addons without the need to modify your installation, but until that time comes (we are working on it) you will need to follow the directions below.
The following has been taken from the above link and updated to PIA 2.8+

Download
The first step is to download the Plugin Architecture. You can get it in either zip or gzip compressed archives.

Extract
You will need to extract this archive. On Windows there are several ways to extract zips/gzips, just use the program of your choice (ex: Winzip). Using Unix you can extract it using a command similar to this one, but your milage may vary depending on the Distribution you are running.
tar -zvxf cacti-plugin-arch.tar.gz
If all goes well, you should have a folder called cacti-plugin-arch with a few patch files and a folders in it. It does not matter in particular to where you download and extract these files to, as we will be moving just the files we need.

Installing

There are two ways of install the Plugin Architecture. The first way is by using the patch files. A patch file contains the difference between the original files and the “new” files, which makes them very small as they only contain exactly what we need to make the changes. The other way is by using the pre-patched full files. These files are the full install of the necessary files with the patch already applied to them. With these you can directly override the files already in your Cacti directory. I only include the files that are necessary to change, so you don't have to override every file in your Cacti install.

Using the Pre-Patched Files

Using the pre-patched files is easiest and most straight forward way to install the Plugin Architecture. You will of course want to backup your Cacti install first before attempting any add-on modifications. Once you have backed up your install. Goto the directory that you extracted the Plugin Architecture to. In this directory you will find several other directories. One of them will look like this *“files-0.8.7g”*. This is to show you that these are the pre-patched files for Cacti v0.8.7g, there may be other versions available if that is not your version.

Now you will need to determine where your original Cacti install is. For instance on Fedora Core 3, my original Cacti files are located at*”/var/www/html/”*. This will vary between Distributions and of course between Linux and Windows, and it is outside the scope of this document to discover where your installation is placed. Once you have the location, remember where it is as you will need it shortly.

You will now copy the files from the *“files-0.8.7g”* directory to your Cacti install directory, overriding any files if you are prompted. There are several ways to copy the files over (Explorer in Windows, FTP, command line, etc…) so I will not go into that here.

From here you are done installing the Plugin Architecture, but it is necessary to configure it first before you continue using Cacti (or Cacti will probably not function properly!)

Using the Patch

Using the patch files is slightly harder that using the pre-patched files, but it is recommended for anyone that has already modified their Cacti install using other mods, or their own custom tweaks. This is mostly used on Linux/Unix etc… but can also be done on Windows if you have the appropriate tools installed. You will ofcourse want to backup your Cacti install first before attempting any add-on modifications.

Now you will need to determine where your original Cacti install is. For instance on Fedora Core 3, my original Cacti files are located at*”/var/www/html/”*. This will vary between Distributions and ofcourse between Linux and Windows, and it is outside the scope of this document to discover where your installation is placed. Once you have the location, remember where it is as you will need it shortly.

Now goto the directory that you extracted the Plugin Architecture to. In this directory you will find several files with names similar to this*"cacti-plugin-0.8.6f.diff"*. This is a patch file that contains everything you need to install the Plugin Architecture. You will copy the file that corresponds with your Cacti version to the location of your Cacti install using a command prompt (if you weren't already using one!)
We will first run this command from the Cacti Install directory

```
patch -p1 -N --dry-run < cacti-plugin-arch.diff
```

This will not make any changes, it will only attempt to do the install and report back any errors. If you receive and FAILED errors, then you know that you will run into a few problems. These problems can usually be addressed by posting in the forums. If you have not modified your Cacti install by using any other mods, then it is usually fairly safe to override the file that "FAILED" with a pre-patched file that is also provided in the archive (See the directions above). Your config.php file will almost always fail to be patched if you have either already configured your database settings for Cacti, or you are using an RPM/DEB install (and possibly even the Windows MSI install). If so, then just override the file, and reconfigure it for your database.
To continue with the patching process, just run this command

```
patch -p1 -N < cacti-plugin-arch.diff
```

This will modify the files and report back and errors. Assuming that all went well, then you can now proceed to configuring your Cacti install.

Configuration
Edit "include/config.php" and specify the database type, name, host, user and password for your Cacti configuration.

```
$database_type = "mysql";
$database_default = "cacti";
$database_hostname = "localhost";
$database_username = "cactiuser";
$database_password = "cactiuser";

/* load up old style plugins here */
$plugins = array();
//$plugins[] = 'thold';

/*
```

```
  Edit this to point to the default URL of your Cacti install
  ex: if your cacti install as at http://serverip/cacti/ this
  would be set to /cacti/
*/
$url_path = "/cacti/";

/* Default session name - Session name must contain alpha characters */

#$cacti_session_name = "Cacti";
```

The $plugins array is required for using the Plugin Architecture (PIA) only. For legacy plugins, those that must be installed in global.php, we have moved the plugins array out of global.php and into config.php. This was done to insure that global.php remains pristine. It is a file that is not intended to be modified, so with the PIA installed, you should not have to.
The variable “$url_path” has also been moved from global.php to config.php for the same reason as the plugins array. For those of you upgrading from very old Plugin Architectures you should know that in the past, we attempted to “detect” this path. However, the process was not reliable. Therefore, you have to specify that path in config.php.
If your Cacti Install is at http://servername/projects/cacti/testing/ then you would need to set the option to this

```
$url_path = "/projects/cacti/testing/";
```

It is important to note that you must include the '/' at the front and end of the location. This is to prevent other issues later down the road.

SQL
The plugin architecture includes a pa.sql file. You will need to import this into your cacti SQL database.
Once this is done, you will have successfully completed installing the Plugin Architecture. You will now want to proceed with downloading and install Plugins. You can refer to Installing Plugins to help with it.