Olli Ruotsalainen

Linux-järjestelmien keskitetty hallinta

NIS-, NFS- ja Puppet-palveluiden avulla

Metropolia Ammattikorkeakoulu
Insinööri (AMK)
Tietotekniikka
Insinöörityö
23.4.2012

Tekijä(t) Otsikko Sivumäärä Aika	Olli Ruotsalainen Linux-järjestelmien keskitetty hallinta NIS-, NFS- ja Puppet-palveluiden avulla 42 sivua + 4 liitettä 23.4.2012
Tutkinto	insinööri (AMK)
Koulutusohjelma	tietotekniikka
Suuntautumisvaihtoehto	tietoverkot
Ohjaaja(t)	yliopettaja Janne Salonen

Tässä insinöörityössä tarkoituksena oli muodostaa ympäristö, jossa Linux-järjestelmiä (työasemia ja palvelimia) hallitaan keskitetysti NIS-, NFS- ja Puppet-palveluiden avulla. Tässä ympäristössä monia järjestelmiä voidaan hallita yhden ainoan järjestelmän avulla. Työssä keskityttiin tarkemmin hallitsemaan enemmän työasemia kuin palvelimia. Lisäksi työssä pyrittiin helpottamaan työasemien asennukseen ja käyttöönottoon liittyviä toimenpiteitä automatisoinnin avulla.

Aluksi työssä tutustutaan keskitettyyn hallintaan yleisesti sekä vertaillaan Windows- ja Linux-järjestelmille saatavia keskitetyn hallinnan palveluita. Tärkeimpänä Windows-pohjaisena vertailukohtana on Microsoft Active Directory, joka tunnetaan suosituimpana Windows-pohjaisena keskitetyn hallinnan palveluna. Sen jälkeen tutustutaan työssä muodostetun keskitetyn hallinnan verkon asennukseen, käyttöön, ylläpitoon ja kehittämiseen. Lopuksi käydään läpi työn tulokset sekä arvioidaan työssä muodostetun keskitetyn hallinnan verkon käytettävyyttä ja tulevaisuutta.

Työn käytännön osuus suoritettiin, koska haluttiin tietää, kuinka keskitetty hallinta Linux-järjestelmillä käytännössä toimii. Tämä osuus onnistui täysin suunnitelmien mukaisesti ja työn lopputuloksiin oltiin tyytyväisiä. Lopputuloksena saavutettiin verkko, jossa monia järjestelmiä voitiin hallita pelkästään yhden järjestelmän avulla. Lisäksi työaseman asennus ja käyttöönotto helpottui huomattavasti automatisoinnin avulla. Näiden saavutusten ansiosta järjestelmäylläpitäjän työtä tehostettiin huomattavasti.

Avainsanat	NIS, NFS, Puppet, Linux, hallinta, AD

Author(s) Title Number of Pages Date	Olli Ruotsalainen Centralized Management of Linux Based Systems Using NIS, NFS and Puppet 42 pages + 4 appendices 23 April 2012
Degree	Bachelor of Engineering
Degree Programme	Degree Programme in Information Technology
Specialisation option	Telecommunications and Data Networks
Instructor(s)	Janne Salonen, Principal Lecturer

The objective of this thesis was to create a Linux based network environment which will be managed centrally using multiple services. These services are NIS, NFS and Puppet. With these services many systems, i.e. workstations and servers, can be managed with one system instead of managing all systems separately. The main focus of this study lies with workstation centralized management. A secondary aim of this study was to automate the installation steps of the workstation.

This thesis has been divided into three sections. The first section is about centralized management in general and what centralized management services are available for Linux based systems. Also in this section centralized management services for Windows and Linux based systems are compared. In this comparison Microsoft Active Directory is in the spotlight for Windows based centralized management.

The second section describes how a Linux based centrally managed network environment can be implemented. The main tasks of the implementation are installing, using, maintaining and improving the network environment.

The last section describes how the objectives were achieved. Also the usability of the current environment is considered.

The reason why the network environment was created for this thesis was to see how centralized management actually works. This study was successful in creating a centrally managed network which is able to manage multiple systems with just one system. Also the automation of the workstation installation turned out to be beneficial. Thanks to the centralized management and installation step automation the system administrator is now able to perform his work much more efficiently.

Keywords	NIS, NFS, Puppet, Linux, management, AD

Sisällys

Lyhenteet

BIOS	Basic Input-Output System, tietokoneohjelma, lataa käyttöjärjestelmän tietokoneen keskusmuistiin sekä käynnistää sen tietokoneen käynnistyessä.
DHCP	Dynamic Host Configuration Protocol, verkkoprotokolla, jakaa automaattisesti IP-osoitteita verkkoon kytkeytyville järjestelmille ja laitteille.
DNS	Domain Name System, nimipalvelujärjestelmä, muuntaa numeromuodossa olevat IP-osoitteet helpommin hahmotettavaan tekstimuotoon.
GPL	General Public License, vapaiden ohjelmistojen julkaisemiseen tarkoitettu lisenssi.
HTTPS	Hypertext Transfer Protocol Secure, tiedon suojattuun siirtoon verkossa tarkoitettu protokolla.
IETF	Internet Engineering Task Force, Internetiin liittyvien asioiden standardoinnista vastaava organisaatio.
IP	Internet Protocol, tärkein Internetin ja eri verkkojen tiedonsiirtoon käytetty protokolla.
LDAP	Lightweight Directory Access Protocol, yleisin hakemistopalveluissa käytetty protokolla.
LTS	Long Term Support, Ubuntu-käyttöjärjestelmän pitkään tuettavissa julkaisuissa käytetty lisätermi.
NAT	Network Access Translation, osoitteenmuutostekniikka, jonka avulla julkista IP-osoitetta voi käyttää useampi verkkoa käyttävä laite.
NFS	Network File System, Sun Microsystemsin kehittämä verkkotiedostojärjestelmä.

NIS Network Information Service, asiakas-palvelin hakemistopalveluprotokolla Unix-pohjaisille järjestelmille.

OpenLDAP Open Lightweight Directory Access Protocol, avoimen lähdekoodin toteutus LDAP-hakemistopalveluprotokollasta.

RAID Redundant Array of Independent Disks, levypakkatekniikka, jonka avulla useat erilliset kiintolevyt yhdistetään yhdeksi loogiseksi levyksi.

RPC Remote Procedure Call, protokolla, joka välittää verkon yli TCP:n avulla proseduurien kutsuja.

SMB Server Message Block, tiedostojen jakamiseen verkon yli tarkoitettu protokolla.

SQL Structured Query Language, tietokantojen hallintaan käytettävä ohjelmointikieli.

SSH Secure Shell, tietoliikenteen salaukseen käytetty protokolla.

SSL Secure Sockets Layer, tietoliikenteen salaukseen käytetty protokolla, varsinkin Web-pohjaisessa liikenteessä.

TCP Transmission Control Protocol, tietoliikenneprotokolla, jolla luodaan yhteyksiä järjestelmien välille.

XML Extensible Markup Language, merkintäkieli, jonka sääntöjen avulla erilaiset dokumentit saadaan luettavaan muotoon. Sitä käytetään formaattina tiedonvälitykseen sekä dokumenttien tallentamiseen.

XML-RPC RPC-protokolla, joka käyttää koodaukseen XML-merkintäkieltä.

YP Sun Yellow Pages, aikaisempi nimi NIS-hakemistopalvelulle.

1 Johdanto

Järjestelmien (palvelimien ja työasemien) määrän kasvaessa organisaation verkossa niiden hallintaan vaadittava työ kasvaa huomattavasti. Tästä seurauksena järjestelmäylläpitäjien on nähtävä enemmän aikaa ja vaivaa ylläpitäessään verkon järjestelmiä. Samalla järjestelmien eroavaisuudet alkavat kasvaa, niiden sisältämä informaatio alkaa muuttua epäorganisoiduksi. Tähän ratkaisuna on suunnitella ja toteuttaa järjestelmien keskitetty hallinta, jonka avulla järjestelmiä voidaan hallita yhdellä järjestelmällä monen sijasta.

Tämän insinöörityön tavoitteena on tutustua Linux-järjestelmien keskitetyn hallinnan palveluiden toimintaan, asennukseen, käyttöön ja ylläpitoon. Työ painottuu enemmän työasemien kuin palvelinten keskitettyyn hallintaan. Työssä myös hieman vertaillaan Linux-järjestelmille saatavia palveluita Microsoftin tarjoamaan Active Directory -palveluun. Lisäksi työn lopussa pyritään automatisoimaan keskitetyn hallinnan verkossa sijaitsevan työaseman asennukseen liittyvät toimenpiteet, jonka avulla työasemien käyttöönotosta tulee entistä helpompaa.

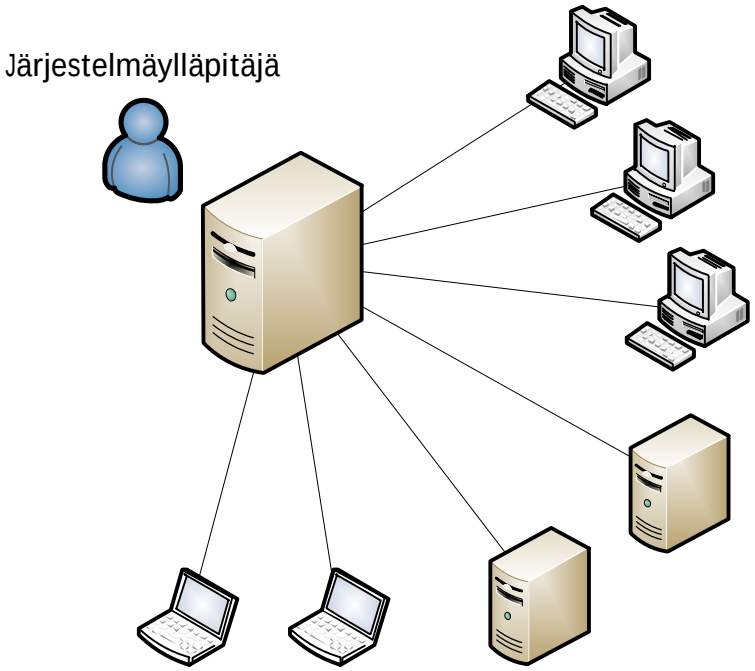
Työ koostuu kolmesta osasta: teoriaosuudesta, käytännönosuuden raportista sekä lopputuloksista. Teoriaosuudessa käydään läpi yleisesti, mitä tarkoittaa keskitetty hallinta ja mitä keskitetyn hallinnan esimerkkipalveluita Linux-järjestelmille on olemassa. Lisäksi näitä palveluita vertaillaan Microsoftin tarjoamaan Active Directory -palveluun. Käytännönosuuden raportissa kerrotaan keskitetyn hallinnan verkon asentamisesta ja ylläpitämisestä Linux-ympäristössä. Työn lopputuloksissa käydään läpi, kuinka käytännön toteutus onnistui sekä arvioidaan työssä muodostetun keskitetyn hallinnan verkon kannattavuutta ja tulevaisuutta.

Työ on tärkeä osa Metropolia Ammattikorkeakoulun tietotekniikan insinöörityötkintoa. Työ toteutettiin itsenäisesti Metropolia Ammattikorkeakoululle, työn ohjaajana toimi yliopettaja Janne Salonen.

2 Järjestelmien keskitetty hallinta

2.1 Yleisesti

Järjestelmien keskitetyllä hallinnalla tarkoitetaan sitä, että esimerkiksi yrityksen verkossa olevia järjestelmiä ei tarvitse hallita ja ylläpitää jokaista erikseen, vaan tämä voidaan hoitaa yhden järjestelmän avulla (ks. kuva 1). Näin ollen muuttuvaa tietoa ei tarvitse muokata kuin yhteen järjestelmään monen järjestelmän sijasta. Tämän ansiosta järjestelmäylläpitäjät säästävät huomattavasti aikaa ja vaivaa verkon ylläpitämisessä. Sen avulla kasvatetaan myös verkon järjestelmällisyyttä, kun tiedetään esimerkiksi, että kaikilla verkon työasemilla on käytössään samat asetukset, palvelut, käyttäjät ja niin edelleen. [3, s. 6, 17; 2.]



Kuva 1. Järjestelmäylläpitäjän hallitsema keskitetyn hallinnan verkko.

Tavallisimpia asioita, joita keskitetysti hallitaan ovat: [1, s. 10-11.]

- käyttäjätunnukset
- salasanat
- verkkonimet
- ryhmämääritykset

- käytäntömääritykset
- tiedostojaot
- sovellukset
- sovellus- ja järjestelmäasetukset
- sovelluspäivitykset
- järjestelmäpäivitykset
- lokitiedostojen hallinta.

Näiden asioiden hallitsemiseen on olemassa erilaisia sovelluksia ja palveluita. Taulukossa 1 lista palveluista ja niitä vastaavista esimerkeistä Windows- ja Linux-järjestelmille.

Taulukko 1. Esimerkkejä keskitetyn hallinnan palveluista Windows- ja Linux-järjestelmille. [29.]

Palvelu	Windows-palvelu	Linux-järjestelmän palvelu
Käyttäjätunnukset Salasanat Ryhmämääritykset Käytäntömääritykset Verkkonimet	Active Directory + Group Policy	Network Information Service (NIS)
Sovellukset	Active Directory + Group Policy	Puppet
Tiedostojaot	Server Message Block (SMB)	Network File System (NFS)
Verkkonimet	Domain Name System (DNS)	Domain Name System (DNS)
Sovellus- ja järjes- telmäasetukset	Active Directory + Group Policy	Puppet

Taulukosta 1 voidaan havainnoida, että Windows- ja Linux-järjestelmien hallinta keskitetysti eroaa toisistaan muun muassa sillä, että Windows-järjestelmissä Microsoft tarjoaa suurimmalta osin omat tuotteensa keskitettyyn hallintaan. Linux-järjestelmissä keskitetty hallinta on toteutettava erillisillä tuotteilla. Molemmille järjestelmille on kuitenkin olemassa vielä vaihtoehtoisia ratkaisuja taulukon 1 esimerkkien lisäksi. Näin ollen Win-

dows-järjestelmillekin saadaan keskitetyn hallinnan palvelut vaihtoehtoisesti toteutettua muilla kuin Microsoftin tarjoamilla tuotteilla. [29.]

2.2 Windows- vai Linux-pohjainen ympäristö

Etuna Microsoftin tarjoamissa tuotteissa kuitenkin ovat, että niiden tuki toistensa kanssa on korkeaa luokkaa. Palveluita ja sovelluksia pystytään helposti yhdistämään ja tästä seurauksena esimerkiksi Active Directory -palvelua saadaan laajennettua yhdistämällä erinäisiä Microsoftin tuotteita keskenään. Keskitetyn hallinnan palveluiden saavuttaminen Linux-ympäristöissä on taattua, mutta palveluiden erinäisyyksien johdosta Active Directoryn kaltaista keskitetyn hallinnan kokonaisuutta ei voida saavuttaa yhtä helposti Linux-järjestelmille kuin Windows-järjestelmille. [29.]

Keskitetyn järjestelmien hallinnan työkaluissa Linux-järjestelmille on etuna se, että ne ovat hyvin useasti täysin ilmaisia käyttää. Taulukossa 1 mainitut NIS-, NFS- ja Puppet-palvelut ovat hyviä esimerkkejä täysin ilmaisista palveluista yksityis- sekä yrityskäyttöön. Hyvä esimerkki kokonaisuudessaan täysin ilmaisesta keskitetyn hallinnan verkosta on Ubuntu Linux -käyttöjärjestelmillä varustetut järjestelmät sekä taulukossa 1 mainitut Linux-pohjaiset palvelut. Tällä verkon kokoonpanolla saavutetaan täysin ilmainen keskitetyn hallinnan ympäristö.

2.3 Hyödyt

Keskitetyn hallinnan tärkeimmät hyödyt verkon ylläpitäjille ovat työn tehostuminen sekä verkon järjestelmällisen rakenteen kasvu. Järjestelmäylläpitäjien työ tehostuu huomattavasti, kun järjestelmiin kohdistuvat ylläpitotoimenpiteet voidaan monen järjestelmän sijasta tehdä yhden järjestelmän avulla. Järjestelmille voidaan suorittaa hyvinkin monimutkaisia ylläpitotoimenpiteitä pienellä vaivalla. Itse järjestelmällisyyttä kasvattaa se, että verkon resurssien kuten verkkonimien, aluenimien ja käyttäjätietojen hallinnasta tulee huomattavasti organisoidumpaa. [3, s. 6, 17; 2.]

Loppukäyttäjä eli organisaation työasemalla työskentelevä käyttäjä hyötyy esimerksiksi keskitetystä käyttäjätunnusten hallinnasta sillä, että hän pystyy kirjautumaan omilla tunnuksillaan mille tahansa verkossa olevalle työasemalle. Samalla hänen henkilökoh-

taiset tiedostonsa (kotihakemisto) ovat aina saatavilla työasemasta riippumattomana. Keskitetyn hallinnan avulla myös järjestelmiin kohdistuvat ylläpitotoimenpiteet voidaan suorittaa siten, että esimerkiksi työasemalla työskentelevä loppukäyttäjä ei näitä millään tavalla havaitse. Näin ollen työasemaa ylläpidetään samaan aikaan, kun loppukäyttäjä työskentelee työaseman äärellä. [5, s. 31; 6, s. 4-5.]

Hallinnan avulla, joka tapahtuu keskitetysti, hyödytään lisäksi myös sillä, että manuaalisen hallinnan virheiden määrä vähentyy huomattavasti. Eli esimerkiksi moneen järjestelmään kohdistuvaa muutosta ei tarvitse jokaiseen järjestelmään tehdä manuaalisesti erikseen. Tämä voidaan tehdä yhdellä järjestelmällä, jolla se kohdistetaan kaikkiin muihin järjestelmiin. Tämä tekee keskitetystä hallinnasta huomattavasti normaalia hallintaa luotettavampaa. Luotettavuutta kasvatetaan myös sillä, että keskitetyssä hallinnassa säästetty aika voidaan käyttää esimerkiksi järjestelmään tehtyjen muutosten tarkistamiseen. [3, s. 18.]

Keskitetty hallinta parantaa myös verkon tietoturvaa, kun tiedetään, missä järjestelmissä toimivat mitkäkin asiat. [3, s. 6.] Tietoturvaa lisää myös se, että kaikki verkon tunnukset kootaan yhteen järjestelmään, joten ylimääräisiä käyttäjiä verkkoon ei pääse ilman oikeanmukaisia tunnuksia. [3, s. 200-201.]

3 Linux-järjestelmän keskitetyn hallinnan palveluita

3.1 Network Information Service (NIS)

Network Information Service eli NIS on asiakas-palvelin-mallin hakemistopalveluprotokolla Unix-pohjaisille järjestelmille. Sen kehitti Sun Microsystems 80-luvun puolivälissä helpottamaan Unix-pohjaisten verkkojen ylläpitoa. NIS tunnettiin aikaisemmin nimellä Sun Yellow Pages (YP). Myöhemmin nimi kuitenkin muutettiin NIS:ksi siitä syystä, että British Telecom -yrityksellä oli jo käytössään tuote nimeltä Yellow Pages. [8; 5, s. 26.]

NIS:n avulla voidaan hallita ja ylläpitää keskitetysti organisaation käyttäjien käyttäjätietoja sekä järjestelmien verkkonimiä. Loppukäyttäjille tämä tarkoittaa sitä, että he voivat käyttää omilla tunnuksillaan kaikkia organisaation työasemia, joihin NIS-palvelu on asennettu ja sallittu heitä varten. NIS-palvelussa ylläpidettäviä käyttäjätietoja ovat

käyttäjätunnukset, tunnusten salasanat, kotihakemistot, ryhmätiedot ja käyttäjien oikeudet. Palvelun ideana on, että kaikki käyttäjätietoihin liittyvät muutokset tehdään vain yhdessä järjestelmässä, josta tieto sitten jaetaan verkon yli muille järjestelmille, joko NIS-asiakkaille tai -sivupalvelimille. [5, s. 31.]

NIS on käyttötarkoitukseltaan tarkoitettu käytettäväksi verkoissa, missä käyttäjien ja työasemien määrä nousee niin suureksi, että näiden yksittäinen hallitseminen käy työlääksi. Esimerkkitilanteena voidaan ajatella organisaatiota, jossa kasvua tapahtuu käyttäjien, työasemien, ryhmien sekä käyttäjien oikeuksien määrässä. NIS-palvelun avulla näiden kaikkien hallinta saadaan keskitettyä yhdelle palvelimelle, jonne haluttavat muutokset tehdään. Tämän ansiosta käyttäjätietojen hallitseminen ja ylläpitäminen muodostuu järjestelmällisemmäksi ja helpottaa suuresti järjestelmäylläpitäjien työtä. [5, s. 31.]

NIS-verkon perustana toimii NIS domain eli NIS-toimialue, joka määritellään heti palvelua käyttöön ottaessa. NIS-toimialueella ei ole kuitenkaan mitään yhteistä DNS-toimialueen kanssa, vaikka usein suositellaan saman nimen käyttöä molemmissa. NIS-toimialueen muodostavat kaikki ne järjestelmät, jotka on liitetty kyseiseen toimialueeseen ja vaihtavat keskenään NIS-palveluun liittyvää tietoa. Toimialueen nimeksi voidaan antaa haluttu nimi. Jos verkossa on kuitenkin jo muita NIS-toimialueita, niin nimi ei saa olla sama kuin muilla toimialueilla. [5, s. 33, 70-71; 9.]

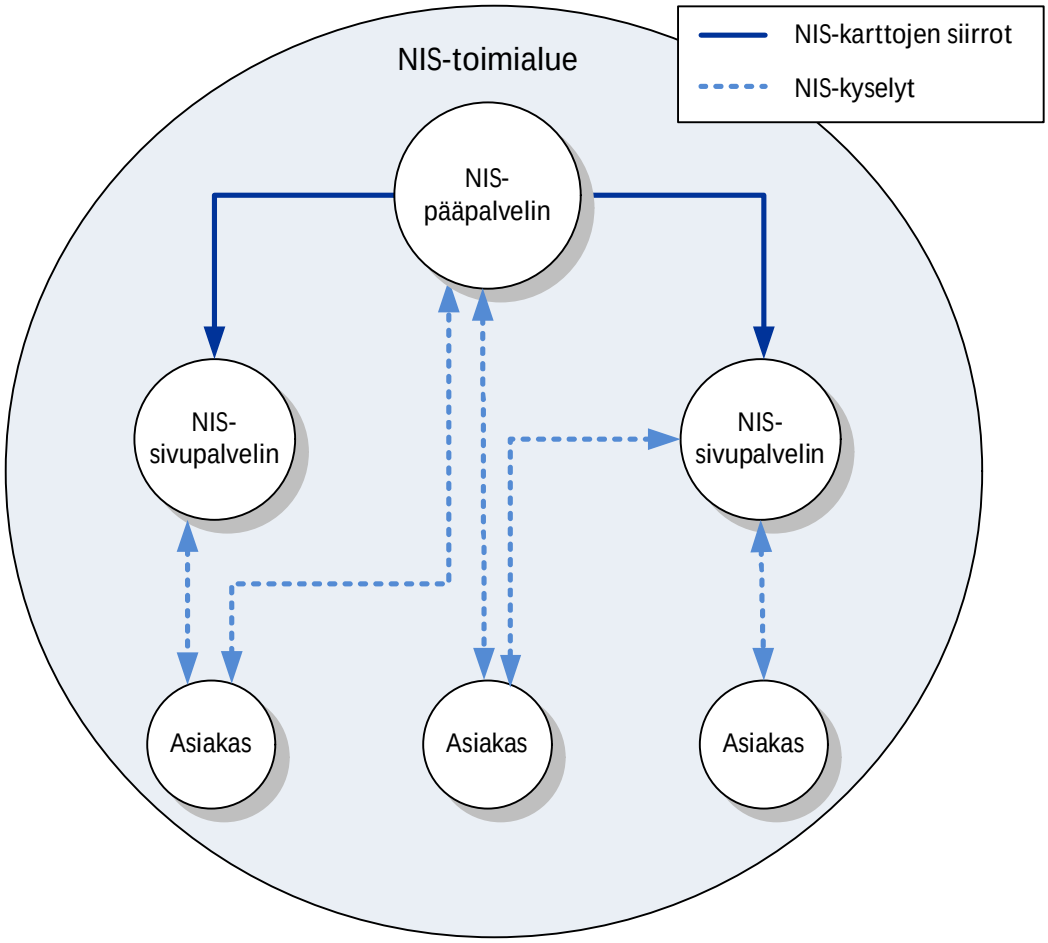
3.1.1 Toiminta

NIS-palvelun toiminta perustuu keskitettyihin käyttäjätietojen tietokantoihin eli karttoihin. Kartat muistuttavat Unix-pohjaisissa järjestelmissä käytettäviä konfiguraatiotiedostoja, joissa tieto on listattu taulukoiden kaltaisesti. NIS:n ylläpitämät kartat riippuvat siitä, minkä Unix-järjestelmän päällä NIS toimii. Tämä johtuu siitä, että käyttäjätietojen tiedostot ja niiden riippuvuussuhteet eroavat hieman toisistaan järjestelmästä riippuen. Taulukossa 2 luettelo yleisimmistä NIS-palvelun kartoista. [5, s. 45-48.]

Taulukko 2. Yleisimmät NIS-palvelun kartat. [9.]

Tiedosto	Kartta / Kartat	Kuvaus
/etc/hosts	hosts.byname, hosts.byaddr	Kuvaa IP-osoitteiden ja isäntänimien välisiä suhteita
/etc/networks	networks.byname, networks.byaddr	Kuvaa IP-verkon osoitteiden ja verkkonimien välisiä suhteita
/etc/passwd	passwd.byname, passwd.byuid	Kuvaa kryptattujen salasanojen ja käyttäjätunnusten välisiä suhteita
/etc/group	group.byname, group.bygid	Kuvaa Ryhmä ID:n ja ryhmien välisiä suhteita
/etc/services	services.byname, services.bynumber	Kuvaa palveluiden kuvausta ja palveluiden nimien välisiä suhteita
/etc/rpc	rpc.byname, rpc.bynumber	Kuvaa RPC-palvelunumeroiden ja RPC-palvelunimien välisiä suhteita
/etc/protocols	protocols.byname, protocols.bynumber	Kuvaa protokollien numeroiden ja protokollien nimien välisiä suhteita
/usr/lib/aliases	mail.aliases	Kuvaa sähköpostialiaksien ja sähköpostialiasnimien välisiä suhteita

NIS-palvelun päätehtävänä on mainostaa käyttäjätietokarttoja ympäri verkkoa NIS-asiakkaille. Tämä tapahtuu asiakas-palvelin-mallin mukaisesti. Tässä mallissa verkossa sijaitsee vähintään yksi NIS-palvelin, joka jakaa tietoa vastaanottaville NIS-asiakasjärjestelmille (ks. kuva 2). Näitä ovat esimerkiksi kaikki verkossa olevat Linux-työasemat. [5, s. 31-32.]



Kuva 2. NIS-verkon rakenne ja toiminta. [5, s. 32.]

NIS-asiakasjärjestelmät käyttävät NIS-palvelimilta saatuja karttoja kahdella eri tavalla, kartasta riippuen. Ensimmäisellä tavalla osa asiakasjärjestelmän konfiguraatiotiedostoista korvataan kokonaan NIS-palvelimelta saaduilla konfiguraatiotiedostoilla heti palvelun käynnistyttyä asiakasjärjestelmässä. Korvattavia tiedostoja ovat muun muassa hosts-, protocols- ja rpc-tiedostot. Toisessa tavassa paikallisia konfiguraatiotiedostoja laajennetaan siten, että NIS-palvelu hyödyntää ensin paikallisen konfiguraatiotiedoston tietoa ja sen jälkeen NIS-palvelimen kartoista saatua tietoa. Laajennettavia tiedostoja ovat muun muassa passwd- ja group-tiedostot. [5, s. 41-43.]

3.1.2 Vikasietoisuus

Edellä mainitusta yksinkertaisesta mallista poiketen verkkoon voidaan pääpalvelimen rinnalle ottaa käyttöön myös sivupalvelimia, jotka toimivat niin sanotusti varapalvelimina (ks. kuva 2). Sivupalvelimen käyttöönotto ei kuitenkaan muuta pääpalvelimen perus

toiminnallisuutta, sillä kaikki käyttäjätietojen muutokset tehdään silti vain pääpalvelimelle. Pääpalvelimelle kuitenkin kerrotaan, jos verkossa on olemassa sivupalvelimia. Näin ollen pääpalvelin mainostaa kartoissa tapahtuneet muutokset sivupalvelimille välittömästi. Sivupalvelimen tärkeimpänä tehtävänä on kasvattaa palvelun vikasietoisuutta, mutta sitä voidaan myös käyttää pilkkomaan NIS-toimialue pienempiin, järjestelmällisiin osiin. [5, s. 31-32.]

Sivupalvelimien käyttö mahdollistaa monien aliverkkojen käytön samanaikasesti toisin kuin yhden pääpalvelimen käyttö. Vähintään yhden sivupalvelimen käyttö on erittäin suotavaa sen tarjoaman vikasietoisuuden vuoksi. [10.]

3.1.3 Tietoturva

NIS:n hyvänä puolena pidetään sen helppoa käyttöönottoa, mutta se tuo mukanaan sen, että NIS ei ole tietoturvaltaan parasta luokkaa. Tietoturva voidaan kuitenkin saavuttaa hyväksyttävälle tasolle pitämällä palvelu päivityksistä ajan tasalla, paneutumalla sen tarjoamiin tietoturva-asetuksiin sekä suunitella oikeanlainen salasanapolitiikka. [5, s. 206-208.]

Yksi suurimmista tietoturvaa koskevista ongelmista on se, että NIS-käyttäjiä ei millään tavalla tunnisteta (autentikoida). Käyttäjät tunnistetaan pelkästään oikealla tunnuksella ja salasanalla. Tärkeä seikka tämän parantamiseksi on erotella NIS-palvelimen oma paikallinen salasanatiedosto ja NIS-salasanatiedosto toisistaan kahdeksi eri tiedostoksi. Näin tehdään siitä syystä, että NIS:n ytimeen eli pääpalvelimeen ei päästä käsiksi millään tavalla. Tärkeää on myös toteuttaa ja ylläpitää oikeanlaista salasanapolitiikkaa. Tämä pitää sisällään muun muassa salasanojen muodon, sisällön, pituuden, käyttöiän sekä muut salasanapolitiikkaa koskevat asiat. [5, s. 206-208.]

Vaikka NIS on helppokäyttöinen, se tarjoaa kuitenkin tietoturvaa koskevien asetusten tarkan määrittelemisen. Näitä asetuksia ovat muun muassa sallittujen ja estettyjen verkkojen ja IP-osoitteiden määritteleminen. Nämä määritellään ypserv.securenets-, host.allow- ja host.deny-tiedostoihin. NIS:n turvallisuutta voidaan myös säätää tarkemmin Linuxilla sijaitsevien Iptables- ja Portmap-palveluiden avulla. [5, s. 209; 9.]

Tietoturvaa voidaan parantaa myös luomalla ja määrittelemällä käyttäjät verkkoryhmiin. Verkkoryhmästä puhutaan nimellä netgroup. Netgroupien avulla voidaan käyttäjiä estää tai sallia käyttämään NIS- ja/tai NFS-palveluita. [11.]

3.1.4 NIS+

Sun Microsystems jatkoi NIS-palvelun kehittämistä kehittämällä palvelun nimeltä NIS+ vuonna 1992. Se kehitettiin paikkaamaan normaalia NIS:iä koskevia vajavaisuuksia kuten tietoturvaa ja hierarkittomuutta. Vaikka NIS+ on kehitetty normaalin NIS-palvelun rungosta, ne ovat rakenteeltaan hyvin erilaisia. [5, s. 27.]

Vaikka NIS+ paikkasi sen aikaisen normaalin NIS-palvelun vajavaisuuksia, se ei kuitenkaan jäänyt Sun Microsystemsin tarjoamaksi hallitsevaksi hakemistoprotokollaksi. Syynä tähän ensinnäkin on se, että NIS+ on todettu palvelinpuolella vaikeasti asennettavaksi ja ylläpidettäväksi. Siitä ei ole myöskään tehty ollenkaan täysin toimivaa toteutusta Linux-järjestelmille ja palvelun kehitys on kokonaan lopetettu. Tästä seurauksena Linux-verkkojen ylläpitäjät käyttävät hakemistopalveluna ennemmin tavallista NIS-palvelua. [30; 31.]

3.2 Network File System (NFS)

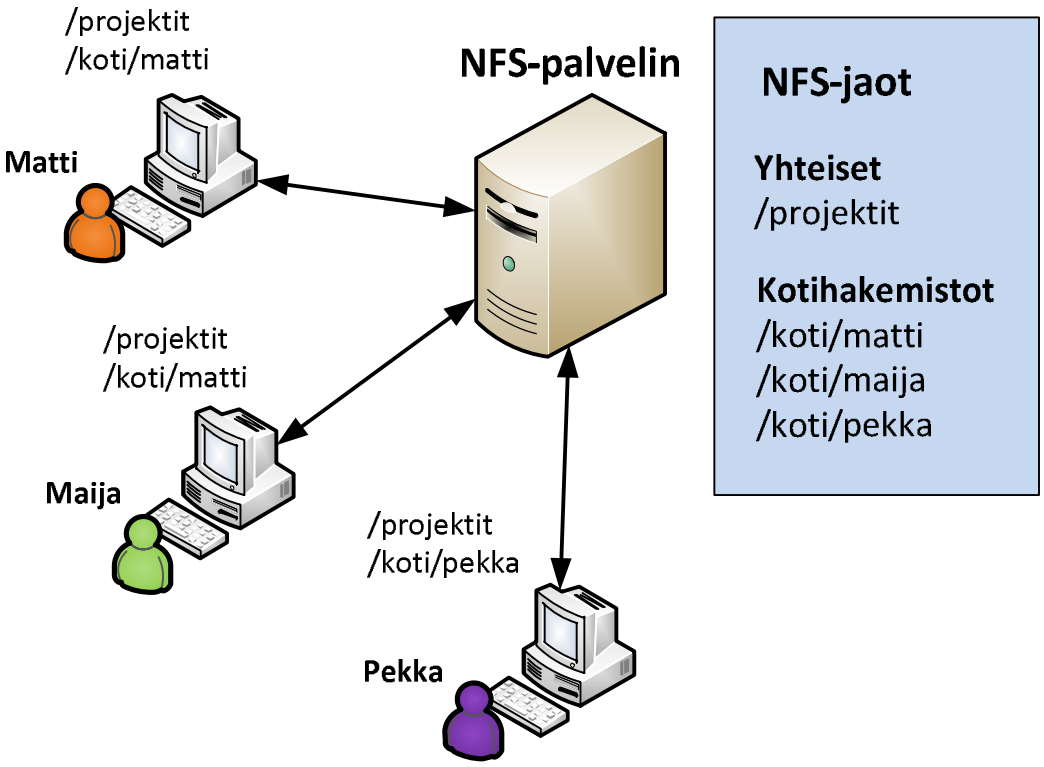
Network File System eli NFS on Sun Microsystemsin on 1980-luvun puolivälin jälkeen kehitetty ja jälkeenpäin IETF:lle laajentunut verkkotiedostojärjestelmä. Se mahdollistaa tiedostojen ja hakemistojen jakamisen verkon yli toisistaan riippumattomien työasemien välillä. Käytännössä tämä tarkoittaa sitä, että etätiedostojärjestelmästä voidaan tehdä paikallinen tiedostojärjestelmä verkon yli ilman, että loppukäyttäjä tätä huomaa. NFS:n avulla tavallisimpia jaettuja asioita ovat kotihakemistot, projektikansiot, levyttömät työasemat ja käyttöjärjestelmien asennukseen tarvittavat tiedostot. [5, s. 9; 12.]

NFS on laajasti käytössä Unix-pohjaisilla järjestelmillä, vaikka se on myös saatavissa Windows-käyttöjärjestelmille. Siitä on olemassa kolme julkaistua versiota: 2, 3 ja 4, joista viimeisin 4.1 julkaistiin vuonna 2010. Versioiden välillä kehitystä on tapahtunut muun muassa tietoturvassa, suorituskyvyssä sekä palvelun keveydessä. [13; 14.]

NFS on suunniteltu käytettäväksi nopeissa verkoissa kuten esimerkiksi organisaation yksityisessä lähiverkossa. Sen käyttöä ei suositella hitailla yhteyksillä, sillä viiveet ja yhteyskatkot kasvavat liian suuriksi ja johtavat NFS:n toimimattomuuteen kyseisissä järjestelmissä. [15.]

3.2.1 Toiminta

NFS toimii samalla palvelin-asiakas-mallilla kuin aikaisemmin mainittu NIS-palvelu. Asiakkaan ja palvelimen välinen tiedonvaihto kulkee Remote Procedure Call (RPC) -protokollan välityksellä. Yksinkertaisessa palvelin-asiakas-mallin verkossa on kuvan 3 mukaisesti yksi NFS-palvelin, joka toimii vain tiedostopalvelimena verkossa oleville NFS-asiakkaille eli työasemille. NFS-palvelin saattaa suurimmassa tapauksessa sisältää esimerkiksi tärkeitä hakemistoja ja tiedostoja, jotka ovat kaikkien työasemien käytettävissä. Näin ollen monet työasemat voivat samanaikaisesti lukea ja kirjoittaa tietoa NFS-palvelimen tarjoamalle tiedostojärjestelmälle. [5, s. 78.]



Kuva 3. NFS-palvelun hakemistojen jakaminen verkon yli.

Yksinkertaisuudesta poiketen verkossa voi olla myös monia erillisiä NFS-palvelimia. Esimerkiksi yhdellä palvelimella voisi sijaita kaikkien käyttäjien kotihakemistot, toisella kaikki projekteihin liittyvät tiedostot ja kolmannella kaikkien laitteiden lokitiedostot. Kaiken tämän lisäksi NFS-palvelu voidaan laajentaa jopa työasemille niin, että ne voivat toimia asiakkuutensa lisäksi myös NFS-palvelimina. Tämän avulla työasemat pystyvät jakamaan paikallisia tiedostoja esimerkiksi toiselle työasemalle.

Palvelimen tarjoama NFS-verkkotiedostojärjestelmä liitetään työaseman paikalliseen tiedostojärjestelmään esimerkiksi suoralla komennolla, automaattisella Automount-liitospalvelulla tai muokkaamalla tiedostoa, johon määritellään käynnistyksen yhteydessä automaattisesti liitettävät tiedostojärjestelmät. Syntaksiin, jolla NFS-liitoksia määritellään, voidaan lisätä muutamia erinäisiä parametrejä, joilla määritellään, kuinka liitos tehdään. Näitä asetuksia ovat muun muassa datagram-koot, liitoksen tyyppi (hard vai soft), aikakatkaisun pituus ja niin edelleen. Liitoksia tehdessä on varmistettava, että hakemisto on olemassa siinä, mihin liitos tehdään. [5, s. 85-87.]

3.2.2 Vikasietoisuus

Jos NFS-palvelulta halutaan vaatia korkeaa saatavuutta ja varmistaa, että verkkotiedostojärjestelmä on aina saatavilla, tulisi verkossa olla vähintään kaksi NFS-palvelinta. Palvelinten tulisi vaihtaa tietoa keskenään synkronisesti, jolloin niin sanottu sivupalvelimen tiedostojärjestelmä olisi aina identtinen pääpalvelimen tiedostojärjestelmän kanssa. Valitettavasti NFS:ssä ei ole suoraa tukea tämän kaltaiselle vikasietoisuudelle. Vikasietoisuus voidaan kuitenkin saavuttaa ottamalla käyttöön GlusterFS-palvelu. Kyseinen palvelu on tarkoitettu käytettäväksi tiedostojärjestelmien jakamiseen palvelinklustereissa. [16, s. 3.] GlusterFS-palvelu pitää huolen, että palvelimet pysyvät tiedostojärjestelmiltään identtisinä, jolloin kaikki pääpalvelimen jaetut tiedostot ja hakemistot ovat tarvittaessa saatavilla myös sivupalvelimella. Jos NFS-pääpalvelin kaatuu, niin vastuu tiedosto- ja hakemistojaoille siirtyy sivupalvelimelle. Näin ollen korkea saatavuus taattaisiin ja NFS-palvelun tarjoamat jaetut tiedostot ja hakemistot olisivat aina saatavilla. [17.]

3.2.3 Tietoturva

Kuten NIS:n, NFS:n peruskäyttöönotto on suhteellisen helppoa, mutta tuo mukanaan pari tietoturvaa koskevaa riskiä. Perusasennukseltaan NFS on oletusarvoisesti yllättävän turvaton ja suojaamaton järjestelmä. Näin ollen sitä tulisi käyttää vain täysin luotetuissa verkoissa eli esimerkiksi yrityksen yksityisessä lähiverkossa. Tietoturvaa saadaan kuitenkin parannettua perehtymällä NFS:ää koskeviin asetuksiin samaan tyyliin kuin aikaisemmin mainitussa NIS:ssä. Näitä ovat sallittujen verkkojen ja IP-osoitteiden määrittäminen, asiakkaan pääsyoikeuksien määrittäminen sekä tiedostojen käyttöoikeuksien määrittäminen. [5, s. 210; 14.]

Sallitut IP-osoitteet ja verkot voidaan määritellä muokkaamalla hosts.deny- ja hosts.allow-tiedostoja. Näihin tiedostoihin voidaan asettaa, mille IP-osoitteille ja verkoille sallitaan pääsy mihinkin palveluun ja miltä IP-osoitteilta se estetään. Rajoituksia tietyille IP-osoitteille ja verkoille voidaan myös tarkemmin määritellä Portmap-palvelun tai Iptables-palomuuripalvelun kautta. Tiedostoja jaettaessa voidaan myös määrittää, kuinka käyttäjät pääsevät jakoihin käsiksi muokkaamalla luku-, kirjoitus- ja root-oikeuksia. NFS:ssä hakemistojen ja tiedostojen omistukset sidotaan käyttäjien ja ryhmien ID:iden mukaan. Omistuksia tehdessä tulisi kuitenkin varmistaa, että palvelimen ja asiakkaan väliset käyttäjä- ja ryhmä-ID:t täsmäävät. [5, s. 217-218; 14.]

3.3 Puppet

Puppet on avoimen lähdekoodin ohjelmisto, jonka avulla voidaan keskitetysti hallita järjestelmien asetustiedostoja ja automatisoida ylläpitoon liittyviä toimenpiteitä. Se on kirjoitettu Ruby-ohjelmointikielellä. Eniten mahdollisuuksia Puppet antaa Unix-pohjaisten järjestelmien ylläpitämiseen, mutta siinä on myös perustason tuki Windows-järjestelmille. [6, s. 3; 18.]

Puppetin kehitti Puppet Labs vuonna 2005, ja sitä julkaistiin aluksi GPLv2-lisenssin alaisuudessa. Myöhemmin versiosta 2.7.0 lähtien sitä on julkaistu Apache-lisenssin alla. Puppetin kehitys aloitettiin, koska markkinoille haluttiin tuoda muita vastaavanlaisia ohjelmia parempi ja kätevämpi työkalu. Sen vahvuudet ovat skaalautuvuudessa, järjestelmäriippumattomuudessa ja ohjelman jatkuvassa kehityksessä. Puppet Labs tarjoaa Puppetin lisäksi maksullista yrityskäyttöön tarkoitettua Puppet Enterprisea, joka on

huomattavasti edistyksellisempi kuin tavallinen Puppet. Ilmainen perustason Puppet on kuitenkin erittäin tehokas työkalu niin koti- kuin yrityskäyttöönkin. [6, s. 3-4; 18.]

Puppetin yksi tärkein vahvuus on, että sen käyttämä kuvaava ohjelmointikieli on selkeää. Tämän ansiosta perusylläpitotoimenpidetiedostojen (manifestien) kirjoittaminen onnistuu helposti eikä vaadi käyttäjältä syvemmän tason osaamista ohjelmoinnista. Puppet on kuitenkin erittäin laaja siitä, että sillä voidaan perus ylläpitotoimenpiteiden lisäksi myös suorittaa erittäin laajoja ja monimutkaisia toimenpiteitä. Kuvassa 4 esimerkki yksinkertaisen manifest-tiedoston sisällöstä. Tämä manifest muuttaa passwd-tiedoston omistuksen root-käyttäjälle, ryhmäomistuksen ryhmälle bin ja tiedoston omistusparametrit muotoon 644. [6, s. 6.]

```
file { "/etc/passwd":
   owner => "root",
   group => "bin",
   mode  => 644,
}
```

Kuva 4. Esimerkki yksinkertaisen manifest-tiedoston sisällöstä. [6, s. 24.]

Puppet käyttää hyväksi Facter-palvelua, jonka avulla saadaan kerättyä tietoa Puppet-asiakkaista. Asiakkaita koskevia tietoja ovat muun muassa asiakkaan käyttöjärjestelmä, laitteisto, IP-osoitteet, MAC-osoitteet, SSH-avaimet sekä paljon muuta. Facterin avulla saatuja tietoja voidaan vahvasti hyödyntää kirjoittaessa manifesteja. Voidaan määrittää esimerkiksi, että kaikki asiakkaat joiden käyttöjärjestelmänä on Ubuntu Linux, tulee asentaa jokin tietty paketti. [6, s. 4, 15, 71-73.]

3.3.1 Resurssit, luokat ja manifestit

Puppetissa ylläpitotoimenpiteitä kutsutaan resursseiksi (resource). Taulukosta 3 nähdään, mitä resursseja Puppetin avulla voidaan hallita. Yhteen tiedostoon voidaan määritellä monta resurssia yhtäaikaa. Tätä kutsutaan resurssien kokoelmaksi (resource collection). Resurssien kokoelmasta puhutaan yleisemmin nimellä luokka (class). Lopuksi nämä luokat voidaan importoida manifest-tiedostoihin, josta ne sitten suoritetaan

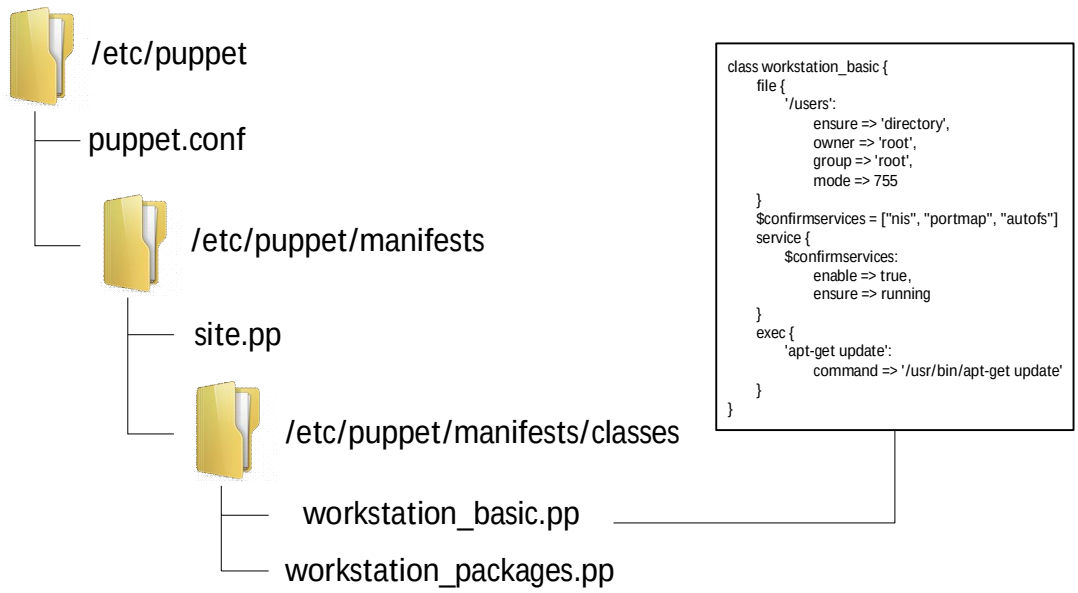
siinä järjestyksessä, missä ne on tiedostoihin määritelty. Puppetissa luokat ja manifestit nimetään tiedostopäätteellä pp. [6, s. 42-61.]

Taulukko 3. Puppet-palvelun tarjoamat ylläpitotoimenpiteet eli resurssit. [6, s. 74-75.]

Resurssin tyyppi	Kuvaus
cron	Cron-ajastuspalveluiden hallinta
exec	Skriptien ajaminen
file	Tiedostojen hallinta
filebucket	Varasto varmuuskopiointia varten
group	Ryhmien hallinta
host	Host-merkintöjen hallinta
interface	Sovittimien hallinta (toimii RedHat ja Solaris Linuxeissa)
mailalias	Sähköposti nimimerkkien hallinta
maillist	Sähköpostilistojen hallinta
mount	Liitospisteiden hallinta
notify	Lähettää viestin puppetd.log-tiedostoon
package	Hallitsee paketteja
schedule	Puppet-ajoitusten määrittäminen
service	Palveluiden hallinta
sshkey	SSH-host-avainten hallinta
tidy	Poistaa turhat tiedostot
user	Käyttäjien hallinta
yumrepo	YUM-repositorioiden hallinta

zones	Solaris-vyöhykkeiden hallinta

Puppet-palvelun tärkein ylläpitotoimenpiteiden tiedosto eli pää-manifest-tiedosto on site.pp, jonka sijainti voidaan nähdä kuvasta 5. Ilman tätä tiedostoa Puppetin ylläpitotoimenpiteitä ei voida suorittaa eikä palvelu toimi. Yleensä tähän tiedostoon tehdään järjestelmäkohtaiset (node-kohtaiset) määritykset, mitä ylläpitotoimenpiteitä kullekin järjestelmälle suoritetaan. Kuvan 5 mukaisessa tapauksessa tiedoston node-kohtaiseen osioon voitaisiin esimerkiksi sisällyttää workstation_basic.pp-tiedosto, josta Puppet-palvelun toimiessa suoritettaisiin kyseisessä tiedostossa luetellut toimenpiteet. [6, s. 24-25, 62, 91-92, 95-98.]

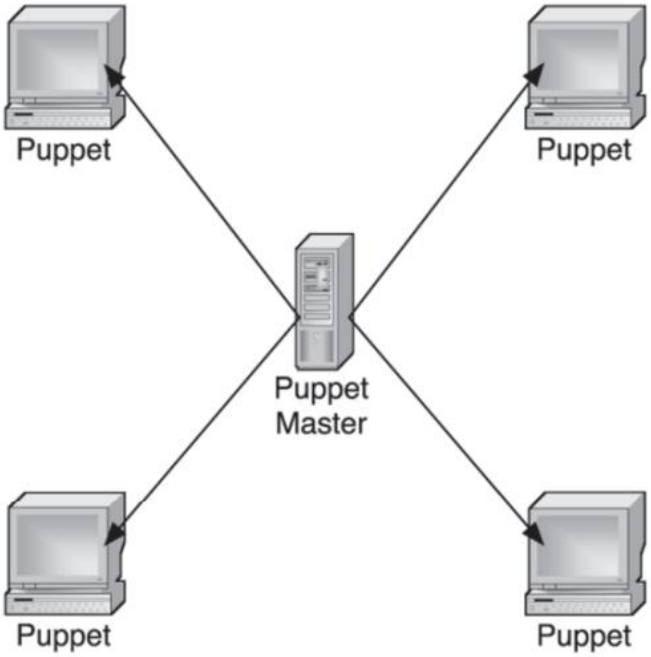


Kuva 5. Esimerkki Puppet-palvelun hakemistorakenteesta. [Apuna 35.]

Puppet-palveluun ei ole pakko luoda monia manifest-tiedostoja. Pelkkä oletuksena luotu site.pp-tiedosto ja sinne kirjoitetut toimenpiteet riittävät. Kuitenkin parhaan käytännön ja selkeyden vuoksi luokat ja manifest-tiedostot olisivat kuitenkin hyvä erotella toisistaan rakentamalla selkeä tiedosto- ja hakemistorakenne. [6, s. 93.]

3.3.2 Toiminta

Puppet toimii asiakas-palvelin-mallin mukaisesti, jossa yksi tai useampi Puppet-palvelin jakaa tietoa asiakkailleen. Puppetissa pääpalvelinta kutsutaan Puppet Masteriksi ja asiakkaita Puppeteiksi (ks. kuva 6). Puppet Masteriin määritellään kaikki ylläpitoa koskevat määritykset, menetelmät, Puppet-asiakkaat sekä muut Puppetia koskevat asetukset. [6, s. 4-5.]



Kuva 6. Puppet-verkon rakenne ja toiminta. [6, s. 5.]

Puppet-asiakkaat eli Puppetit ovat verkon työasemia tai palvelimia, joita halutaan keskitetysti Puppetin avulla hallita (ks. kuva 6). Ne toimivat vain Puppet Masterilta tulevan tiedon vastaanottajina. Puppet-asiakas vastaanottaessaan tietoa se tarkistaa, että Puppet Masterin tarjoama tieto on paikallisesti ajan tasalla. Jos näin on, tieto vain tarkistetaan, mutta mitään ei kuitenkaan suoriteta. Jos kuitenkin jotkin määritellyt asetukset eroavat Puppet Masterin ja Puppetin välillä, asetukset päivitetään Puppetiin ajan tasalle. [6, s. 4-7.]

Puppet Masterin ja Puppetien välinen tiedon vaihto tapahtuu oletuksena puolen tunnin välein. Tämä voidaan kuitenkin halutessa asiakaskohtaisesti muuttaa siten, että se ta-

pahtuu useammin tai harvemmin. Puppet Masterilta voidaan halutessa tehdä myös manuaalinen välittömästi tapahtuva asetusten lähettäminen. [6, s. 4.]

3.3.3 Sertifikaatit

Puppet Masterin avulla hallitaan myös asiakas-palvelin-välisiä sertifikaatteja. Sertifikaattien avulla varmistetaan, että vain halutut asiakkaat voivat vaihtaa tietoa Puppet-palvelimen kanssa. Näin ollen ylimääräisiä asiakkaita ei pääse Puppet-verkkoon ilman Puppet Masterin hyväksyntää. Sertifikaatin hyväksyntä on kaksivaiheinen. Ensin asiakas pyytää Puppet Masterilta sertifikaattia, jonka jälkeen Puppet Master tiedon saadessaan hyväksyy sertifikaatin. Tästä seuraa se, että Puppet-yhteys on muodostettu ja tiedon vaihto voi alkaa. [6, s. 25-27.]

Oletuksena Puppet Masterin ja asiakkaiden välillä on oltava oikeanmukainen signeerattu sertifikaatti, mutta tämä voidaan kuitenkin halutessa kiertää ja käyttää automaattista sertifikaattien signeerausta. Tätä ei kuitenkaan suositella käytettäväksi tietoturvasyistä, koska muuten kuka tahansa samassa verkossa oleva Puppetilla varustettu asiakas olisi mukana Puppet-verkon tiedonvaihdossa. [6, s. 25-27.]

3.3.4 Tietoturva

Puppet on tietoturvaltaan erittäin hyvä, sillä kaikki sen liikenne kulkee kryptatusti palvelimen ja asiakkaiden välillä. Yhteyden luontiin sekä liikenteen kulkemiseen Puppet käyttää XML-RPC-verkkopalveluja, joiden yli yhteys menee turvatun HTTPS:n kautta TCP-portissa 8140. [6, s. 4.]

Toinen vahvaa tietoturvaa takaava ominaisuus ovat jo aikaisemmin mainitut sertifikaatit. Puppetissa nämä palvelimen ja asiakkaiden väliset sertifikaatit ovat standardeja SSL-sertifikaatteja. Sertifikaatit kryptataan ja lopullinen asiakkaan autentikointi tapahtuu manuaalisesti Puppet Masterin kautta. Sertifikaattien ja autentikoinnin avulla varmistetaan, että vain halutut järjestelmät pääsevät Puppet-verkkoon, eivätkä ulkopuoliset järjestelmät pääse kyseiseen verkkoon käsiksi. [6, s. 4.]

3.4 Eroavaisuuksia Active Directory -palvelun kanssa

Suurin ero Active Directory -palvelulla ja aikaisemmin mainituilla Linux-järjestelmien keskitetyn hallinnan palveluilla on NIS-palvelussa. Active Directory käyttää tiedonvaihtoon huomattavasti NIS:iä yleisempää LDAP-protokollaa. Kyseisen protokollan suurin vahvuus on sen toimivuus sekä Windows- että Unix-pohjaisilla järjestelmillä, kun taas NIS toimii vain Unix-pohjaisilla järjestelmillä. LDAP-protokolla on myös uudempi kuin NIS, joten sen tietoturvan taso on nykyaikaisempi tarjoten mahdollisuuden muun muassa helposti toteutettavaan autentikointiin. LDAP-protokollan sisällä liikkuu myös enemmän tietoa kuin NIS-protokollan sisällä, esimerkiksi käyttäjien puhelinnumerot, sähköpostiosoitteet, yhteystiedot ja niin edelleen. [19; 20; 5, s. 28-29.]

Eroavaisuus Active Directory ja Linux-järjestelmän tarjoamien palveluiden välillä on myös niiden tarjoamien ominaisuuksien laajuudessa. Siinä missä Linux-järjestelmillä keskitetty hallinta toimii kolmen eri palvelun alla: NIS hallitsee käyttäjätietoja, NFS hoitaa tiedostojen jakoa ja Puppet hallitsee sovelluksia sekä palveluita koskevia muutoksia, hoituu tämä Windows-järjestelmillä kätevästi Active Directory ja Group Policy -palveluilla. Jotta Linux-järjestelmille saavutettaisiin Active Directory –palvelun kaltainen verkon kokonaisuuden hallintaa tarkoitettu palvelu, tämä tulisi toteuttaa hieman monimutkaisemmin käyttöönottamalla monia yksittäisiä palveluja. [4.]

4 Käyttöönotto

4.1 Suunnitelma

Käyttöönotossa työn tavoitteena on toteuttaa järjestelmien keskitetty hallinta Linux-ympäristössä. Tämän ansiosta saadaan aikaan ympäristö, jonka järjestelmien asennus ja ylläpito käyvät verkon ylläpitäjältä mahdollisimman tehokkaasti. Työssä keskitytään hallitsemaan enemmän työasemiin kuin palvelimiin kohdistuvia asioita. Mitä hallitumpi ja automatisoidumpi työasemien asennuksista ja käyttöönotoista saadaan, sen paremmin tavoitteisiin päästään.

Tarkempana työn tavoitteena on saada NIS-, NFS- ja Puppet-palvelut toimimaan verkossa halutulla tavalla. Näiden palveluiden ansiosta verkossa saadaan toimimaan muun muassa seuraavat asiat:

- käyttäjän kirjautuminen omilla tunnuksilla mille tahansa työasemalle
- käyttäjän henkilökohtaisten tiedostojen (kotihakemisto) saatavuus tiedostonjakopalvelimelta mille tahansa työasemalle
- haluttujen palveluiden ja sovelluksien automaattinen asennus
- haluttujen palveluiden ja prosessien taustalla toimiminen ja niiden varmistaminen
- sovellus- ja järjestelmäasetusten ajan tasalla pysyminen
- pakettivarastolistojen ajan tasalla pysyminen
- uuden työaseman mahdollisimman automatisoitu käyttöönotto.

Koska kyseessä on työ, joka rakennetaan vain testitarkoitukseen, tehdään se virtuaalisesti käyttäen viimeisimpiä virtualisointitekniikoita. Tämän avulla työn toteuttaminen ei vaadi kuin yhden fyysisen järjestelmän, jonka alle luodaan tarvittava määrä virtuaalisia järjestelmiä eli virtuaalikoneita.

4.1.1 Laitteisto

Työhön tarvittavat laitteet tarjoaa Metropolia Ammattikorkeakoulu. Työn toteuttamista varten tarvitaan palvelinta, jossa on riittävästi suorituskykyä muutaman virtuaalikoneen virtuaaliympäristöä varten. Tätä varten palvelimeksi valitaan Dell PowerEdge 2950 (ks. kuva 7).

Kuva 7. Dell PowerEdge 2950 -palvelin. [34.]

Virtuaaliympäristön muodostamisessa voidaan luottaa Dell PowerEdge 2950 -palvelimeen, sillä laitteistonsa puolesta palvelin sopii erittäin hyvin työn vaatimaan virtuaaliympäristön muodostamiseen. Palvelimen tarkemmat laitetiedot taulukossa 4.

Taulukko 4. Dell PowerEdge -palvelimen laitetiedot.

Palvelimen tyyppi	Räkkipalvelin
Prosessori	Quad-Core Intel Xeon 5320 1.86 GHz
Muisti	8 Gt DDR2, 667 MHz, ECC, Fully Buffered DIMM
RAID	RAID-5, PERC 5/i RAID adapter
Verkkokortti	2 x Ethernet verkkokortti (10/100/1000MB)

Kun palvelin saadaan asennettua ja konfiguroitua toimivaksi virtualisointialustaksi, sen hallitsemista varten tarvitaan erillistä hallintatyöasemaa. Hallintatyöaseman tulee olla samassa koulun yksityisessä laboratorioverkossa kuin virtualisointipalvelimen. Hallintatyöasemaksi asennetaan erillinen virtuaalikone koulun yksityiseen laboratorioverkkoon.

4.1.2 Ohjelmisto

Dell PowerEdge -palvelimeen virtualisointialustaksi valitaan VMware ESXi, jonka erikoisuus on siinä, että se toimii palvelimella omana käyttöjärjestelmänään. Palvelimen hallitsemista varten hallintakoneelle pitää asentaa VMware vSphere Client -sovellus. Sovelluksen avulla luodaan työssä tarvittava virtuaaliympäristö, sen vaatimat virtuaalikoneet sekä hallitaan kaikkea ympäristöön liittyvää.

Suunniteltuun virtuaaliympäristöön tarvitaan järjestelmiksi työasemia sekä palvelimia. Käyttöjärjestelmiksi palvelimille valitaan täysin komentopohjainen Ubuntu Server 10.04 ja työasemille graafinen Ubuntu Desktop 10.04. Uusimpia versioita Ubuntuista ei valita, koska työssä vaadittujen palvelujen tueksi suositellaan kyseistä versiota muun muassa sen LTS-merkistä, joka takaa pitkän tuen kyseiselle versiolle. Hallintatyöasemalle käyttöjärjestelmäksi valitaan Windows 7.

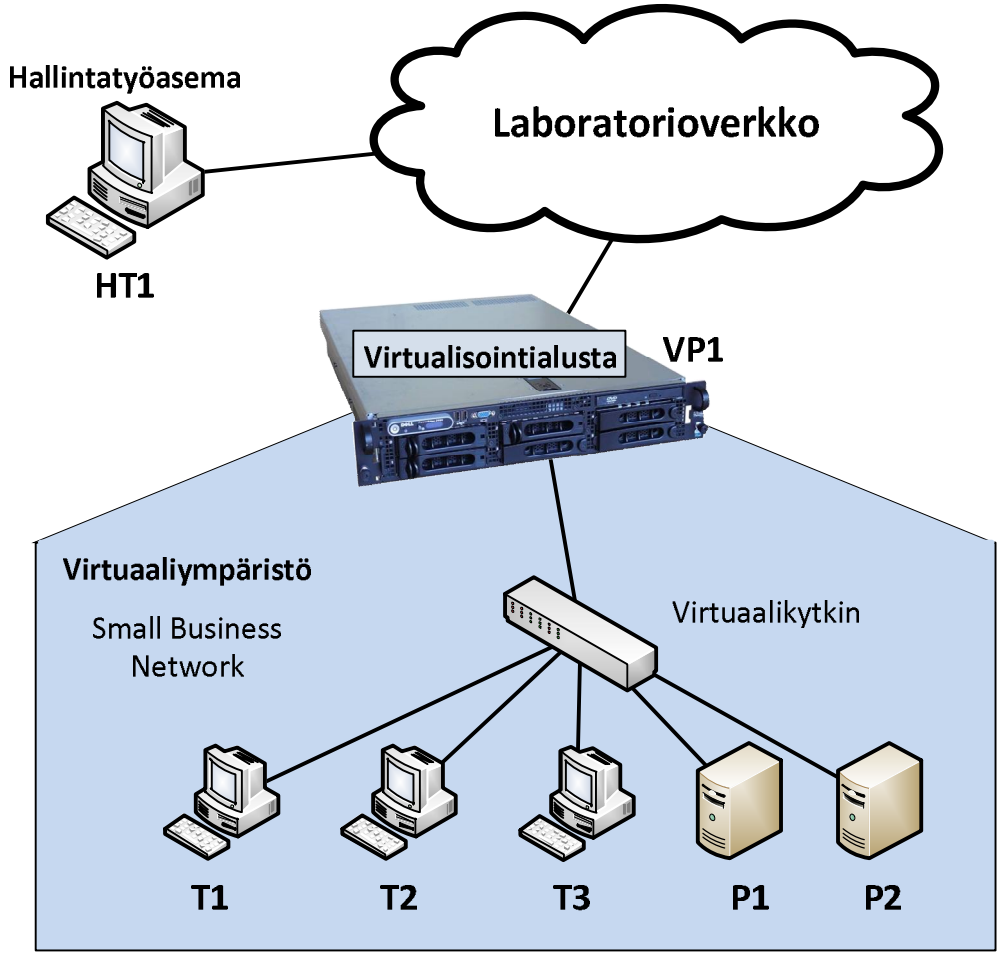
NIS-, NFS- ja Puppet-palveluista asennetaan viimeisimmät saatavilla olevat versiot Ubuntun pakettivarastojen kautta. NFS-palvelu kuitenkin jätetään vain toimivaksi versio 3 -tasolle siitä syystä, että versio 3 sopeutuu helpomman hallittavuutensa ansiosta paremmin kyseiseen työhön. Näiden palveluiden lisäksi työasemille asennettaan viimeisin versio Automount-palvelusta, jonka avulla työasemille saadaan liitettyä automaattisesti tulevia NFS-tiedostojakoja.

Virtuaalikoneille asennetaan edellä mainittujen palveluiden lisäksi OpenSSH-server-palvelu, jonka avulla hallintakoneelta voidaan ottaa tekstipohjainen etäyhteys jokaiselle virtuaalikoneelle. Näin ollen virtuaalikoneita voidaan hallita suoraan käyttäen VMware vSphere Client -sovelluksen graafista hallintakonsolia tai SSH:n yli käyttäen PuTTY-etähallintasovellusta.

Hallintakoneen ollessa virtuaalinen siihen otetaan etäyhteys joko yhdestä koulun fyysisestä Windows-käyttöjärjestelmällä varustetusta työasemasta tai kotikoneelta SSH-tunnelin avulla. Hallintakoneen etäkäyttöä varten käytetään Windows-käyttöjärjestelmän tarjoamaa Etätyöpöytä-sovellusta sekä tunnelointia varten PuTTY-etähallintasovellusta.

4.1.3 Topologia

Verkon topologia muodostuu fyysisistä sekä virtuaalisista laitteista. Topologiakuvan (ks. kuva 8) mukaisesti järjestelmät nimetään loogisesti niin, että T-kirjaimen omaavat nimet (HT1 ,T1, T2 ja T3) tarkoittavat työasemia ja P-kirjaimen omaavat nimet (VP1, P1, P2) tarkoittavat palvelimia. Järjestelmien host-nimet ja tarkemmat tiedot löytyvät taulukosta 5.



Kuva 8. Työtä varten muodostettu verkkotopologia.

Verkon tärkeimpänä elementtinä toimii virtualisointipalvelin VP1 (ks. kuva 8), jonka sisälle virtuaalikoneet rakennetaan. Virtuaalikoneet liitetään virtuaaliympäristössä toimivaan virtuaalikytkimeen (ks. kuva 8), jonka ansiosta tuleva ja lähtevä liikenne kulkeutuu virtuaaliympäristössä oikeille järjestelmille.

Taulukko 5. Verkkotopologiassa kuvassa 8 olevat laitteet ja niiden tiedot.

Nimi	Host-nimi	IP-osoite	Verkko	Kuvaus
HT1	Win7RDC	10.95.250.79	10.95.250.0	Hallintatyöasema
VP1	dellpe	10.95.202.100	10.95.202.0	Virtualisointialusta
T1	test-workstation-desktop	10.95.202.101	10.95.202.0	Työasema testausta varten
T2	workstation2-desktop	10.95.202.102	10.95.202.0	Työasema
T3	workstation3-desktop	10.95.202.103	10.95.202.0	Työasema
P1	nfs	10.95.202.252	10.95.202.0	NFS-palvelin
P2	puppetnis	10.95.202.253	10.95.202.0	NIS- ja Puppet-palvelin

Kaikki virtuaaliympäristössä olevat järjestelmät liitetään samaan 10.95.202.0-verkkoon, jossa sijaitsee myös itse VP1-virtualisointipalvelin. Palvelimille P1 ja P2 annetaan verkon kaksi viimeistä vapaata IP-osoitetta. Osoitetta 10.95.202.254 ei voida käyttää, koska se on määritelty laboratiorioverkon kytkimelle yhdyskäytäväosoitteeksi, jota käytetään myös virtuaalikoneiden yhdyskäytävänä. Virtuaalisten työasemien staattisten IP-osoitteiden annossa lähdetään liikkeelle osoitteesta 101 alkaen. Hallintatyöaseman IP-osoitteena käytetään koulun tarjoamaa vapaata IP-osoitetta. Järjestelmien nimipalvelimina käytetään koulun laboratorioverkon nimipalvelimia osoitteissa 10.95.254.252 ja 10.95.254.253.

4.2 Toteutus

Työn toteutuksessa edettiin porrasperiaatteella. Tämä tarkoittaa sitä, että tehtävästä A ei siirrytty tehtävään B ennen kuin tehtävä A on täysin valmis. Tätä varten laadittiin tehtävälista (taulukko 6), jonka mukaan työssä edettiin. Tämän avulla työ eteni järjestelmällisesti, ja työnteko oli tehokasta.

Taulukko 6. Työssä suoritettavat tehtävät.

Työn numero	Työn kuvaus
1.	Laitteistoon tutustuminen ja fyysiset laitteistoasennukset
2.	VMware ESXi -virtualisointialustan asennus ja konfigurointi
3.	Virtuaalikoneiden asennus ja valmiuteen konfigurointi
4.	Etäyhteyksien konfigurointi ja niiden testaus
5.	NIS-palvelun asennus ja testaus
6.	NFS-palvelun asennus ja testaus
7.	Puppet-palvelun asennus ja testaus
8.	Edellä mainittujen palveluiden yhdistys ja testaus
9.	Edellä mainittujen palveluiden ylläpito ja työasemien testaus
10.	Työn varmistus ja lopullinen testaus. Jos aikaa jää, niin työn kehittäminen

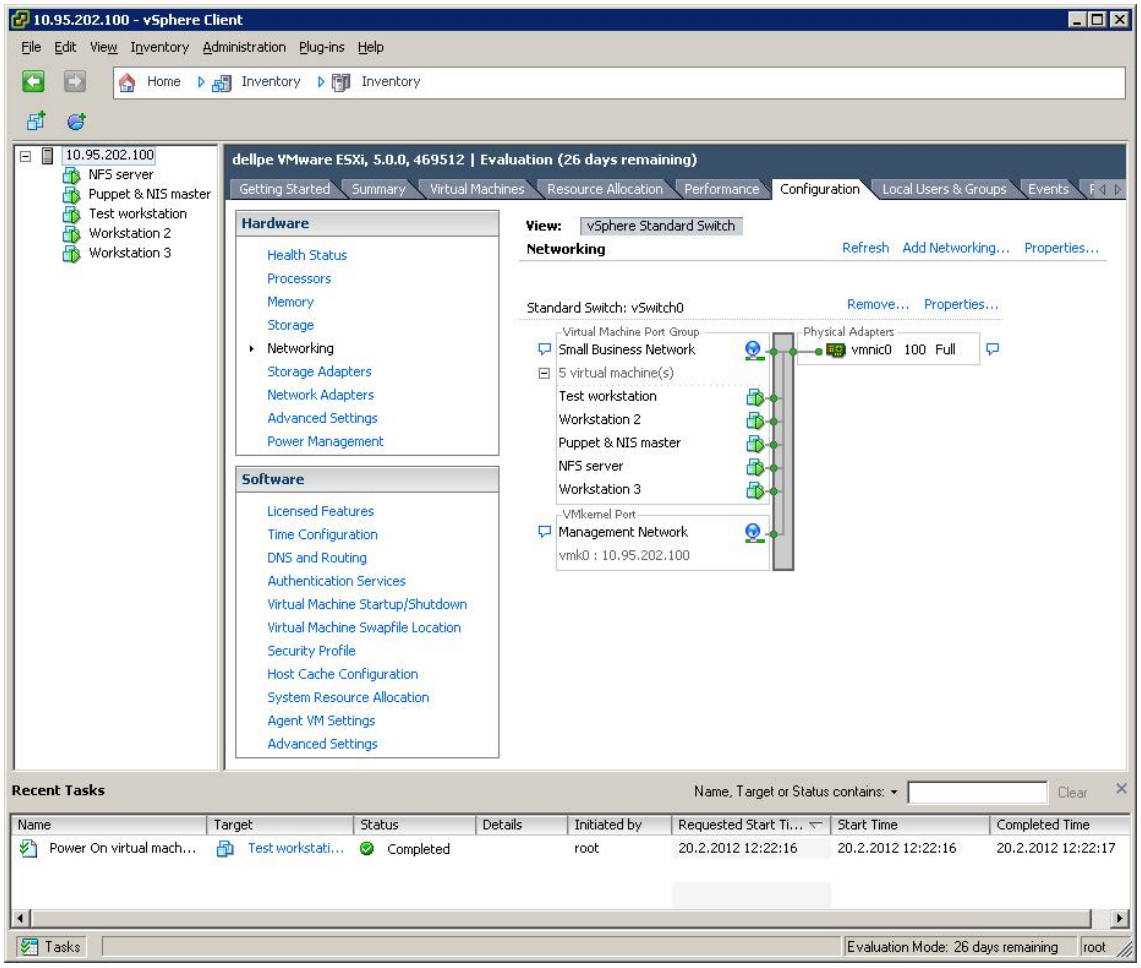
Tehokkuuteen vaikuttavana asiana voidaan mainita ongelmatilanteiden järjestelmällinen ratkonta, sillä vikoja etsiessä voitiin keskittyä siihen, mitä muutoksia kyseisellä niin sanotulla portaalla oli tehty. Jos jokin lakkasi toimimasta, niin voitiin aina palata taaksepäin perumalla esimerkiksi järjestelmään viimeksi tehdyt muutokset.

Työssä korostettiin porrasetenemistä ja ongelmatilanteiden hallintaa sillä, että hyväksi käytettiin VMware-virtualisointialustan snapshot-ominaisuutta. Tällä ominaisuudellla virtuaalikoneista voidaan tallentaa niin sanottuja tilakatsauksia, joilla virtuaalikone voidaan myöhemmin tarvittaessa palauttaa tuohon tilaan, jolloin snapshot otettiin. Tätä ei kuitenkaan suositella käytettäväksi liian paljon, sillä snapshot-tilojen tallentaminen vaatii virtualisointialustalta jonkin verran ylimääräistä levytilaa. [28.]

4.2.1 Alustus

Työ aloitettiin Dell PowerEdge 2950 -palvelimen siirtämisellä koulun laboratorioluokan palvelinkaappiin. Seuraavaksi jatkettiin palvelimen alustamisella ja asetusten säätämisellä. Palvelimen BIOS:sta varmistettiin, että laite on kunnossa ja samalla kiintolevyt alustettiin. Kiintolevyt asennettiin toimiviksi RAID 5 -tekniikalla.

Palvelin liitettiin koulun laboratorioverkkoon oman 10.95.202.0 255.255.255.0 aliverkkonsa alle. Tämän jälkeen koulun verkkoon luotiin Windows 7 -käyttöjärjestelmällä varustettu virtuaalinen hallintakone palvelimen hallintaa varten. Hallintakoneelle asennettiin VMware vSphere Client -sovellus virtuaaliympäristön hallintaa varten (ks. kuva 9).



Kuva 9. VMware vSphere Client -hallintasovellus, virtuaaliverkko ja virtuaaliset järjestelmät.

Hallintasovelluksen avulla aloitettiin virtuaalikoneiden asentaminen. Virtuaalikoneita luotiin 2 palvelimen sekä 3 työaseman verran (ks. kuva 9). Palvelimille asennettiin Ubuntu Server -käyttöjärjestelmät ja työasemille asennettiin Ubuntu Desktop -käyttöjärjestelmät. Puhtaiden asennuksien jälkeen järjestelmille määriteltiin verkkoasetukset, DNS-nimipalvelimet, päivitettiin pakettivarastot, päivitettiin käyttöjärjestelmät sekä vaihdettiin käyttöjärjestelmän täydelliseksi kieleksi suomi. Helpottaaksemme palvelinten tekstipohjaista hallintaa niihin asennettiin tuki SSH-etäyhteyksille. Viimeisenä alustavana toimenpiteenä testattiin, että jokainen virtuaalikone sai yhteyden verkkoon ja yhteydet toimivat suunnitelmien mukaisesti.

4.2.2 Asennus ja konfigurointi

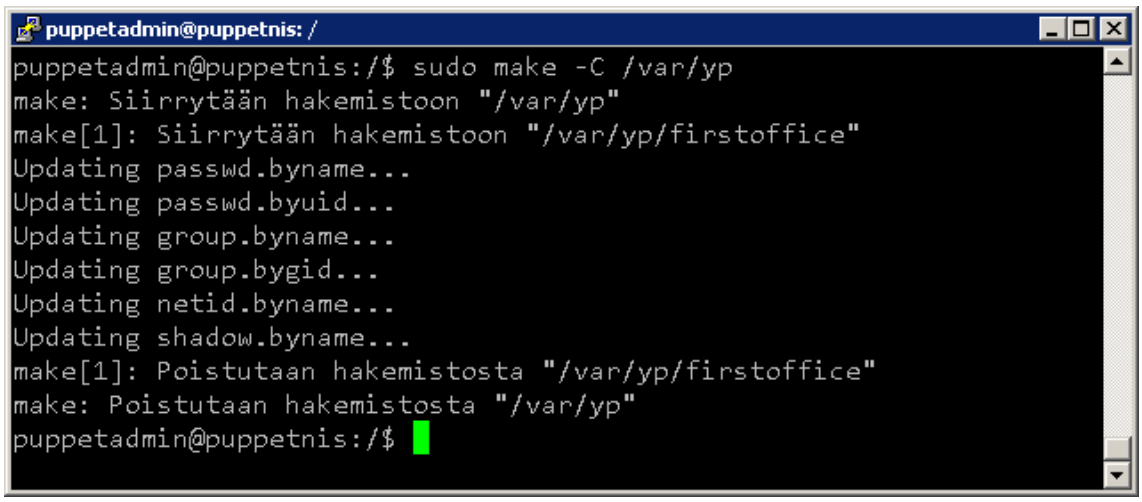
Asennuksissa ja konfiguroinneissa ensisijaisina ohjeina käytettiin Ubuntu Community -yhteisön virallisia HowTo-ohjeita (lähteet 21, 22 ja 24) ja Puppet-kirjaa (lähde 6). Hyvänä apuna toimi myös Ubuntu Suomi -yhteisön keskustelupalstan keskustelu vuodelta 2006 (lähde 23). Pienissä erityyppisissä ongelmatilanteissa ja niiden ratkaisemisissa apuna käytettiin monia erinäisiä Internetissä sijatisevia Linux-aiheisia keskustelupalstoja.

Järjestelmien keskitetyn hallinnan rakentaminen aloitettiin tehtävälistan mukaisesti NIS-palvelun asennuksella ja konfiguroinnilla. Palvelu asennettiin palvelimelle P2, jonka tehtävänä oli toimia tulevana NIS- ja Puppet-palvelimena. Palvelua asentaessa asennukselle määriteltiin tuleva NIS-domain-nimi. Nimeksi päätettiin antaa *firstoffice*.

Jotta NIS-palvelu alkoi toimia halutulla tavalla, muokattiin muutamaa palveluun liittyvää asetusta palvelun asetustiedostojen kautta. Palvelulle kerrottiin, että kyseinen palvelin toimii NIS-pääpalvelimena ja pääpalvelin tunnistaa itsensä toimivaksi nimen *firstoffice* alla. Lisäksi muokattiin tiedostoa, jonka avulla NIS-palvelun tietoturvaa parannettiin sallimalla palveluun vain halutun verkon järjestelmät. Tarkemmat muutokset palvelimen asetuksiin löytyvät työn liitteestä 1.

Asetusten muokkaamisen jälkeen NIS-palvelu käynnistettiin uudestaan, jotta tehdyt muutokset tulivat välittömästi voimaan. Tämän jälkeen palvelimelle luotiin ensimmäistä kertaa palvelun tärkein elementti, NIS-kartat. Karttojen ensimmäinen luonti tapahtui

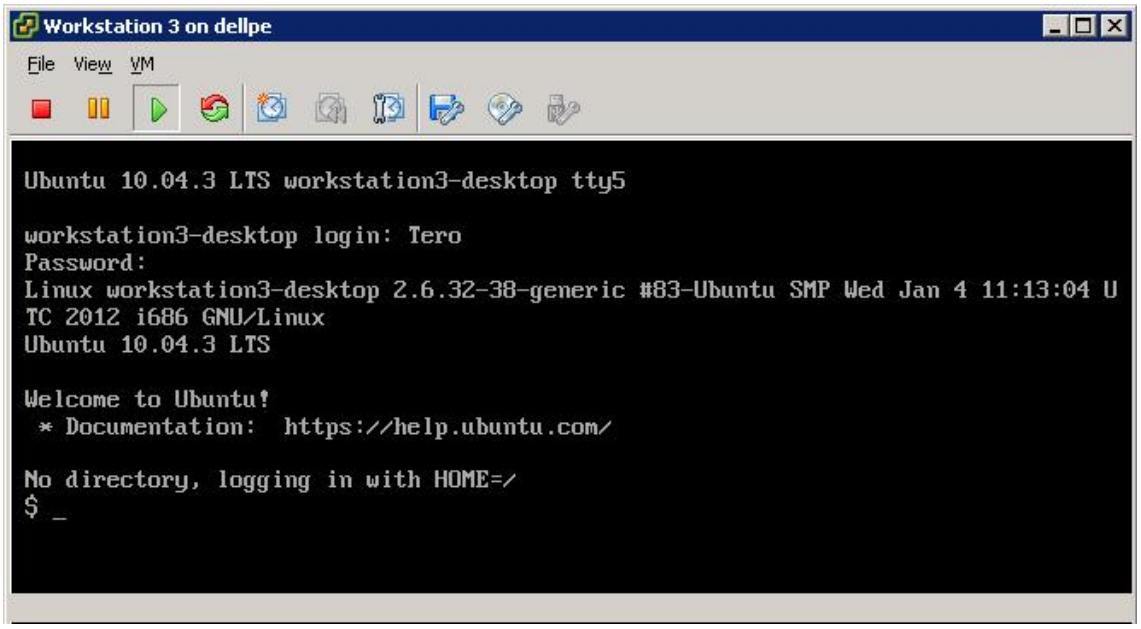
komennolla */usr/lib/yp/ypinit –m*. Kun kartat ensimmäisen kerran luodaan, niitä ei tarvitse luoda enää toista kertaa, sillä tiedon muuttuessa karttoja vain päivitetään.



Kuva 10. NIS-karttojen päivitys suoritettuna onnistuneesti.

Jotta palvelua voitiin testata, järjestelmään luotiin neljä loppukäyttäjää: Matti, Maija, Pekka ja Pirjo. Luonnin yhteydessä käyttäjille määriteltiin myös salasanat. Jotta luodut käyttäjät tulivat voimaan NIS-palveluun, piti ne päivittää luotuihin karttoihin. Tämä tapahtui kuvan 10 mukaisesti komennolla *make –C /var/yp*. Komento päivittää kaikki NIS-palvelun kartat kerralla. Karttoja voidaan myös päivittää tyyppi kerrallaan, mutta tässä työssä kaikki kartat päivitettiin varmuuden vuoksi aina samalla kertaa.

Palvelun testausta varten tarvittiin vähintään yksi NIS-asiakasjärjestelmä. Tätä varten asennettiin NIS-palvelu test-workstation-desktop-työasemalle (T1). Jotta palvelu sai tietää, että sen pitää toimia kyseisessä järjestelmässä asiakkaana, oli tätä varten tehtävä muutama muutos palvelun asetuksiin. Palvelun asetuksiin määriteltiin NIS-domain-nimi, NIS-pääpalvelimen IP-osoite sekä pääpalvelimen host-nimi. Palvelulle kerrottiin myös muutamaa tiedostoa muokkaamalla, että paikallisten käyttäjätietojen puuttuessa lisää käyttäjätietoja löytyy NIS-palvelimen käyttäjätietokartoista. Tarkemmat tiedot asiakkaan asetuksista löytyvät työn liitteestä 1.



Kuva 11. Käyttäjä kirjautuneena onnistuneesti komentopohjaiseen ympäristöön.

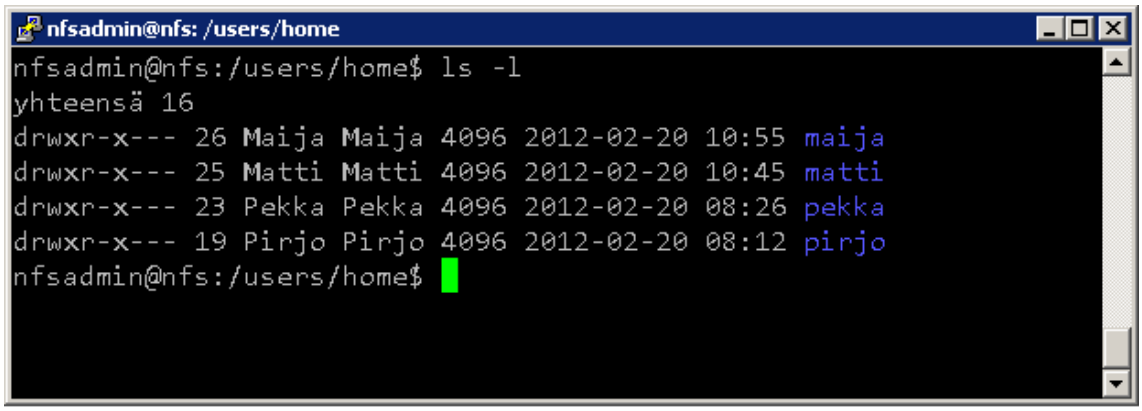
Asetusten määrittelemisen jälkeen asiakkaan NIS-palvelu käynnistettiin uudestaan. Nyt NIS-palvelimelle lisättyjen käyttäjien oli mahdollista kirjautua asiakastyöasemille käyttäen omia tunnuksiaan. Tässä vaiheessa tekstipohjainen kirjautuminen oli vain mahdollista, sillä käyttäjillä ei ollut vielä käytössään keskitettyä kotihakemistoa (ks. kuva 11), joka vaaditaan käyttäessä graafista työpöytäympäristöä. Tämä kuitenkin saavutettiin seuraavaksi pystyttämällä NFS-tiedostojakopalvelu.

Suunnitelman mukaisesti seuraavaksi asennettiin NFS-palvelu sille tarkoitetulle omalle P1-palvelimelleen. NFS-palvelusta on kaksi erillistä pakettia riippuen, onko kyseessä oleva järjestelmä palvelin vai asiakas. Näin ollen palvelimelle asennettiin palvelimelle suunnattu paketti. Tämän jälkeen luotiin hakemistot */users* ja */users/home* tulevia suunniteltuja kotihakemistoja varten. Jotta nämä hakemistot saatiin jaettua verkon yli, muokattiin NFS-jakoja koskevaa asetustiedostoa sekä parannettiin jakojen tietoturvaa sallimalla vain tietyn verkon järjestelmät. NFS-palvelimen asetusten säädöt ja muokkaukset on kerrottu tarkemmin työn liitteessä 2.

NFS-palvelimen ollessa kunnossa verkkoon lisättiin kaksi NFS-asiakasta. Ensimmäiseksi asiakkaaksi lisättiin NIS-palvelin P2. Palvelimesta tehtiin NFS-asiakas asentamalla siihen asiakasta varten asennettava paketti. Sen jälkeen järjestelmään luotiin samat hakemistot kuin NFS-palvelimelle sekä muokattiin myös hieman tietoturva-asetuksia sallimalla

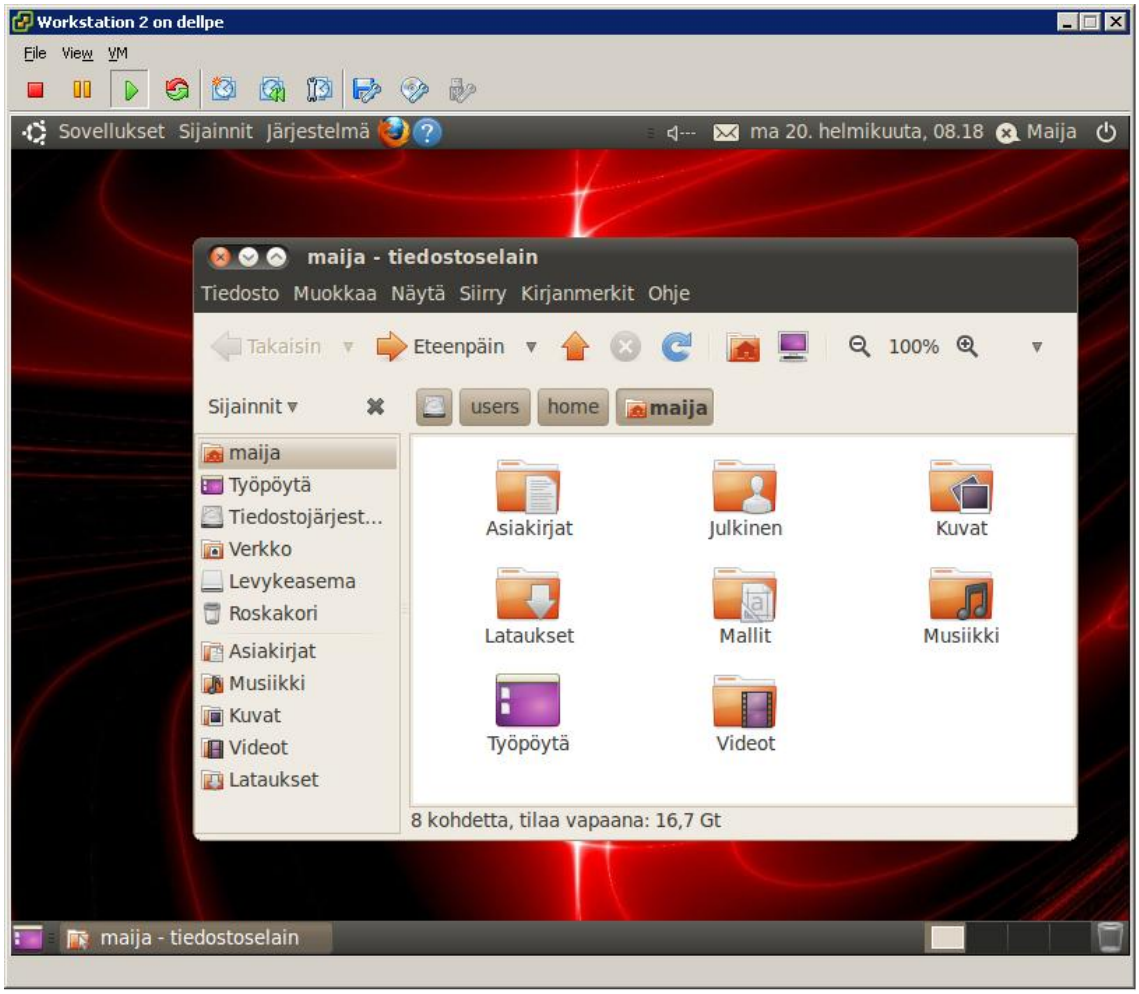
vain NFS-yhteys halutulta palvelimelta. Viimeisenä määriteltiin fstab-tiedoston asetus, jonka avulla NFS-tiedostojärjestelmä liitetään järjestelmään automaattisesti käynnistyksen yhteydessä. NFS-asiakkaan asetusten muokkauksesta on tarkemmin liitteessä 2.

Palvelimesta tehtiin NFS-asiakas siitä syystä, että uusia käyttäjiä luodessa heidän kotihakemistonsa tästä eteenpäin luotiin erilliselle tiedostonjakopalvelimelle P1 eikä paikallisesti omaan järjestelmään. Näin ollen käyttäjien kotihakemistot sijaitsivat tästä lähtien keskitetysti yhdellä palvelimella. Tästä syystä aikasemmin luodut käyttäjät Matti, Maija, Pekka ja Pirjo poistettiin ja luotiin uudestaan, mutta tällä kertaa luonnin yhteydessä heille tehtiin ja määriteltiin kotihakemisto */users/home*-hakemiston alle. Kotihakemistot luotiin oikeuksin 750 eli vain käyttäjä itse pääsee käsiksi omaan kotihakemistoonsa (ks. kuva 12).



Kuva 12. Käyttäjien kotihakemistot NFS-palvelimen /users/home-hakemistossa.

Työasemasta test-workstation-desktop (T1) tehtiin ensimmäinen NFS-työasema-asiakas asentamalla NFS-asiakaspaketti ja muokkaamalla asetukset kuntoon samalla tavoin kuin P1-palvelimella. Ainoana poikkeuksena työaseman fstab-tiedostoa ei muokattu vaan tilalle asennettiin Automount-palvelu. Palvelun avulla ainoastaan käyttäjäkohtainen kotihakemisto liitetään järjestelmään kirjautumisen yhteydessä eli esimerkiksi */users/home/matti* eikä koko hakemistoa */users/home*. Tämä kasvattaa NFS-palvelun käyttöoikeuksien hallintaa sekä tietoturvaa. Näiden jälkeen käyttäjät pystyivät kirjautumaan työaseman kautta graafiseen käyttöliittymään ja heidän kotihakemistonsa olivat heillä käytössään verkon yli NFS-palvelimelta (ks. kuva 13).



Kuva 13. Loppukäyttäjä onnistuneesti kirjautuneena graafiseen työpöytäympäristöön.

Viimeisenä keskitetyn hallinnan palveluna asennettiin Puppet. Palvelimelle tarkoitettu Puppet Master -paketti asennettiin P2-palvelimelle, jossa sijaitsee myös NIS-palvelu. Asennuksen jälkeen muokattiin Puppetin asetustiedostoja. Asetuksissa määriteltiin, että kyseinen palvelin toimii Puppet Master -palvelimena. Asetuksiin määriteltiin myös, että palvelin toimii myös asetustiedostojen jakana. Samassa määriteltiin, että hakemiston */etc/puppet/files* alle sijoitetaan kaikki verkossa jaettavat asetustiedostot. Näiden jälkeen Puppet oli konfiguroitu ja palvelu käynnistettiin uudestaan, jotta asetukset tulivat voimaan. Palvelimen konfiguroinnista tarkemmat tiedot työn liitteessä 3.

Puppet-palvelimen ollessa kunnossa tehtiin työasemasta test-workstation-desktop (T1) sekä NFS-palvelimesta (P1) Puppet-asiakkaita. Järjestelmiin asennettiin Puppet-asiakaspaketit ja muokattiin palvelun asetuksia. Niihin määriteltiin, mikä on Puppet-palvelimen nimi ja mistä osoitteesta palvelin löytyy. Asetuksiin määriteltiin myös, millä

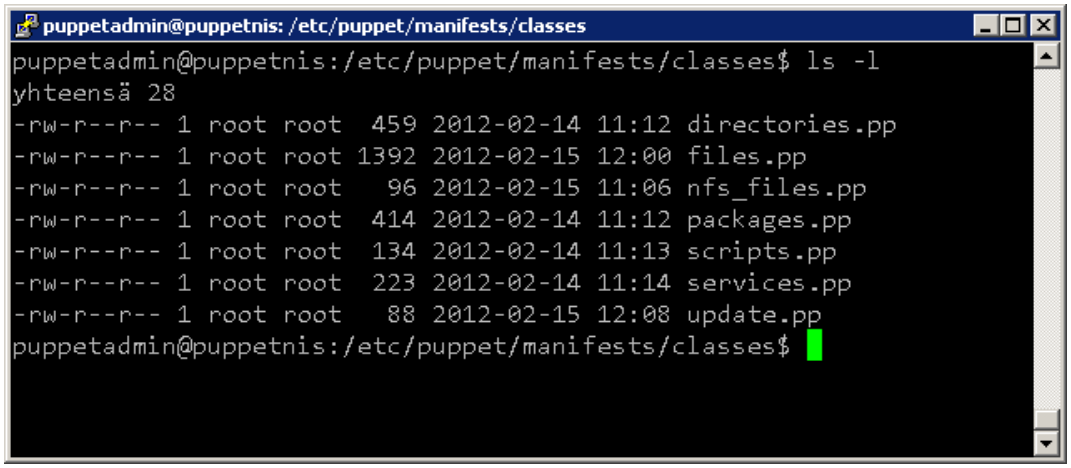
nimellä asiakas sertifioidaan ja kuinka usein palvelu kyselee palvelimelta järjestelmään tehtäviä muutoksia. Näiden jälkeen Puppet-asiakas sertifioitiin ottamalla yhteys palvelimeen testikomennon avulla ja hyväksyttiin asiakas Puppet-palvelimen komentojen avulla.

Kuva 14. Sertifikaatin signeeraus suoritettuna onnistuneesti työasemalle.

Kuvan 14 mukaisen sertifikaattihyväksynnän jälkeen palvelimen ja asiakkaiden välinen Puppet-yhteys saatiin toimimaan. Puppet-palvelimelle ei ole kuitenkaan vielä määritelty minkäänlaisia ylläpitotoimenpiteitä, joten vaikka yhteys toimi, niin mitään ei vielä kuitenkaan tapahtunut. Tarkemmat tiedot Puppet-asiakasta koskevista säädöistä ovat työn liitteessä 3.

4.2.3 Automatisointi

Kun NIS-, NFS- ja Puppet-palvelut saatiin onnistuneesti toimimaan, seuraava askel oli automatisoida kyseisten palveluiden asentaminen ja konfigurointi. Tähän hyödynnettiin viimeisimmäksi asennettua Puppet-palvelua. Sen avulla voitiin automatisoida pakettien asentaminen, palveluiden tilojen määritteleminen, asetustiedostojen siirtäminen, skriptien ajaminen sekä päivitysten asentaminen loppukäyttäjältä piilossa taustalla suoritettuna. Näiden avulla uuden työaseman käyttöönotosta tulisi entistä tehokkaampaa.



Kuva 15. Puppet-palvelun erilliset ylläpitotoimenpidetiedostot eli manifestit.

Puppet-palvelun työasemalla suoritettavat ylläpitotoimenpiteet jaettiin kuuteen eri tiedostoon eli manifestiin. Lisäksi luotiin yksi NFS-palvelimelle tarkoitettu toimenpidetiedosto. Tiedostot eriteltiin sen mukaan, minkälaisia ylläpitotoimenpiteitä eli resursseja tiedosto pitää sisällään. Jokaiselle edellä mainitulle oli kuvan 15 mukaisesti oma tiedostonsa: hakemistojen luonnit, tiedostojen siirrot, pakettien asennukset, palveluiden tilamääritykset, suoritettavat skriptit sekä pakettivarastojen päivitykset.

Tiedostojen siirtoon liittyvään manifest:iin määriteltiin, että Puppetin jaetussa hakemistossa */etc/puppet/files* olevat tiedostot jaetaan muille Puppet-asiakkaille halutulla tavalla. Tässä vaiheessa toimivaa työasemaa test-workstation-desktop (T1) hyödynnettiin siirtämällä oikein säädetyt NIS-, NFS- ja Automount-palveluiden asetustiedostot edellä mainittuun Puppet-palvelimen tiedostonjakohakemistoon. Tiedostojen siirto työasemalta palvelimelle tehtiin SSH-yhteyden yli. Tämän jälkeen palveluiden toimivat asetustiedostot (ks. taulukko 7) voitiin jakaa verkon yli halutuille Puppet-asiakkaille.

Taulukko 7. Puppet-palvelimen jaettavat eri palveluiden asetustiedostot.

Tiedosto	Palvelu	Kuvaus
defaultdomain	NIS	NIS-domain-nimen asetus
nsswitch.conf	NIS	Käyttäjätietojen etsimistä koskevat asetukset
yp.conf	NIS	NIS-pääpalvelimen sijantia koskevat asetukset

yp_add.sh	NIS	Skripti käyttäjätietotiedostojen asettamiseen
hosts.allow	NFS	Sallittujen yhteyksien määritykset
hosts.deny	NFS	Estettyjen yhteyksien määritykset
auto.home	Automount	Jaettujen kotikansioiden liitosasetukset
auto.master	Automount	Paikallisten liitoshakemistojen asetukset
nfs_motd.tail	Järjestelmä	Uusi message-of-the-day-tiedosto NFS-palvelimelle

Jotta kaikki manifestit eli ylläpitotoimenpiteet suoritettaisiin uuden asiakkaan liittyessä ja sertifioituessa Puppet-verkkoon, oli työasemalle tehtävä määritys päämanifestitiedostoon site.pp. Tiedostoon kirjoitettiin node-osiot jokaiselle työasemalle erikseen, johon sitten luokat sijoitettiin mukaan. Myös NFS-palvelinta varten tehtiin oma osio, jossa olivat toimenpiteet: pakettivarastojen päivitys ja Message-of-the-day (MOTD) -tiedoston siirto.

Puppet-palvelun toimiessa halutulla tavalla, uusien työasemien workstation2- (T2) ja workstation3-desktop (T3) käyttöönotto helpottui huomattavasti. Kun työasemille oli asennettu puhdas Ubuntu-käyttöjärjestelmä, laitettu verkkoasetukset kuntoon, asennettu Puppet-asiakassovellus ja sertifioitu Puppet-yhteys, niin kaikki muu tapahtui Puppet-palvelun ansiosta järjestelmässä automaattisesti. NIS-, NFS-, ja Automount-palvelut asennettiin, palveluiden asetukset konfiguroitiin, palvelut aktivoitiin, liitoshakemistot luotiin, tarvittavat skriptit ajettiin sekä pakettivarastot päivitettiin. Työasemien uudelleen käynnistyksen jälkeen halutut toimenpiteet olivat suoritettu onnistuneesti ja työasemat olivat valmiita loppukäyttäjän käytettäväksi. Myös NFS-palvelimella suoritettiin onnistuneesti sille määritellyt ylläpitotoimenpiteet, jotka olivat pakettivarastojen päivitys sekä MOTD-tiedoston muokkaus.

4.2.4 Testaus ja ylläpito

Työssä muodostettua keskitetyn hallinnan ympäristöä testattiin loppukäyttäjien näkökulmasta monilla tavoilla. Esimerkiksi kaikille työasemille kirjauduttiin yhtäaikaisesti eri

loppukäyttäjillä ja työasemia käytettiin yhtäaikasesti luomalla tiedostoja henkilökohtaiseen muilta piilossa olevaan kotihakemistoon. Erinäisten testien jälkeen havaittiin, että ympäristö toimi onnistuneesti ja työasemien käyttö toimi moitteetta.

Työssä muodostettua keskitetyn hallinnan verkkoa voidaan kyseisillä palveluilla ylläpitää monella tapaa. Nämä ylläpitoa koskevat huomiot tehtiin niiden kokemusten pohjalta, mitä ympäristön työasemien ja palvelinten käytössä sekä testauksessa ilmeni.

Kun verkkoon haluttiin luoda, poistaa tai muokata uusia käyttäjiä tai ryhmiä, nämä kaikki toimenpiteet voitiin suorittaa NIS-palvelun avulla. Oli kuitenkin muistettava, että palvelun käyttäjätietokarttoja pitää ylläpitää ja päivittää aina käyttäjätietoja muokatessa siitä syystä, ettei verkossa ilmene ongelmia käyttäjätunnusten käytössä. Verkon koon ja käyttäjätilien kasvaessa olisi suositeltavaa luoda automaattisesti esimerkiksi kaksi kertaa päivässä ajettava skripti varmuuden vuoksi siitä syystä, että käyttäjien käyttäjätiedot pysyisivät aina mahdollisimman ajan tasalla.

Käyttäjien ylläpidon lisäksi verkon NFS-tiedostojakoja voitiin myös ylläpitää yhden järjestelmän avulla. Jos verkkoon haluttiin jaettavaksi enemmän hakemistoja ja/tai tiedostoja, voitiin toimenpide suorittaa joko suoraan NFS-palvelimen jakoja koskeviin asetuksiin tai sitten lähettää halutut muutokset asetustiedostoina Puppet Master -palvelimelta asiakkaana toimivalle NFS-palvelimelle. Verkon kaikkiin työasemiin voitiin esimerkiksi ottaa käyttöön yhteinen projektihakemisto, jonne käyttäjät pystyisivät päivittämään projekteja koskevia tiedostoja. Tämän toteuttaminen oli mahdollista parin asetustiedoston muokkaamisen avulla, ja se voitiin toteuttaa keskitetysti Puppet-palvelua apuna käyttäen.

Keskitetyn hallinnan avulla myös sovelluksien ja pakettien asentaminen saatiin toimimaan muutamalla tiedoston muokkauksella. Jos esimerkiksi kaikkiin työasemiin haluttiin asentaa esimerkkinä toimisto-ohjelmisto LibreOffice 3.0, niin tämä onnistui helposti vain lisäämällä pari riviä tekstiä Puppet-palvelun ylläpitotoimenpidetiedostoihin. Sovellus asentui taustalla automaattisesti jokaiseen työasemaan aikaisintaan puolen tunnin kuluttua tai viimeistään seuraavalla työaseman käynnistyskerralla. Puppet-palvelun avulla voidaan automatisoida tarvittaessa hyvinkin monen tyyppisiä ylläpitotoimenpiteitä. Nämä tyypit on listattu työn alussa mainitussa taulukossa 3.

Puppet-palveluun voidaan myös määritellä hyvin tarkasti, mille järjestelmille tehdään mitäkin ylläpitotoimenpiteitä. Puppet-palvelimen site.pp-tiedostoon, johon määritellään ylläpitotoimenpiteet jäjestelmäkohtaisesti, voidaan tehdä hyvinkin monipuolisia määrityksiä. Aikaisemmin mainittuna voidaan esimerkiksi määrittää, että kaikkiin verkon järjestelmiin, joissa on käyttöjärjestelmänä palvelin- tai työasemaversio Ubuntu 10.04, asennetaan jokin tietty sovellus. Tässä työssä järjestelmät pyrittiin määrittelemään Puppet-palveluun mahdollisimman yksinkertaisesti.

4.2.5 Kehittäminen

Vaikka työn tavoitteisiin oli jo päästy, niin työn kehitystä pohdittiin vielä automatisoinnin lisäämisellä, luotettavuuden kasvattamisella, vikasietoisuuden kehittämisellä sekä Puppet-palvelun raportointiominaisuuden hyödyntämisellä. Kuitenkin vain automatisoinnin lisäämistä ehdittiin käytännössä toteuttaa. Muut kehittämiseen liittyvät asiat jätettiin suosiolla vain pohdiskelujen varaan.

Työasemien asennukseen liittyviä toimenpiteitä voidaan kätevästi automatisoida vieläkin enemmän esimerkiksi komentosarjan eli skriptin avulla. Työaseman asennusta tehostettiin kirjoittamalla skripti, joka voitiin ajaa heti puhtaan Ubuntu-käyttöjärjestelmäasennuksen jälkeen. Lyhyesti sanottuna skripti asettaa työasemalle halutun 10.95.202.0-verkon IP-osoitteen, määrittää DNS-nimipalvelinten osoitteet, päivittää pakettivarastot, asentaa uusimmat päivitykset, asentaa Puppet-palvelun ja konfiguroi Puppet-palvelun. Skriptin tarkempi sisältö löytyy työn liitteestä 4. Tämän jälkeen työasemien ylläpitäjälle ei jäänyt muita toimenpiteitä, kuin sertifioida Puppet-asiakas palvelimen kautta. Sertifioinninkin olisi saanut automaattisesti toimimaan, mutta tietoturvasyiden takia yhteyksien varmistus oli hyvä jättää manuaaliseksi toimenpiteeksi.

Kehitettyä automatisointia kokeiltiin onnistuneesti palauttamalla työasema workstation3-desktop (T3) snapshot -ominaisuuden avulla tilanteeseen, missä siihen oli asennettu vain puhdas käyttöjärjestelmä eikä mitään muuta. Skriptissä määritellyt toimenpiteet suoritettiin onnistuneesti ja skripti toimi halutulla tavalla. Enää työasema tarvitsi vain sertifioida, jonka jälkeen Puppet-palvelussa määritellyt toimenpiteet suoritettiin ja tästä seurauksena työasemasta tuli vieläkin pienemmällä vaivalla käyttövalmis loppukäyttäjää varten.

Keskitetyn hallinnan verkossa, jossa sijaitsee eri palveluita tarjoavia palvelimia, tulisi miettiä näiden luotettavuutta. Verkkoa voidaan kehittää kasvattamalla verkon vikasietoisuutta. Helpoimmin tämä onnistuu lisäämällä verkkoon sivupalvelimia pääpalvelimien rinnalle. Sivupalvelimien tehtävänä on varmistaa, että pääpalvelimien ollessa alhaalla, haluttu tieto on saatavilla sivupalvelimella. Näin vältyttäisiin verkossa esiintyviltä erinäisiltä katkoksilta mahdollisimman hyvin. Työssä käytetyt NIS-, NFS- ja Puppet-palvelut jo aikasemmin mainittuna tukevat jokainen kyseistä pää- ja sivupalvelimien mallia. Kaiken lisäksi sivupalvelinten asentaminen ja käyttöönotto ei ole tehtävänä vaikea toteuttaa, mutta vaatii kuitenkin perehtymistä palveluiden asetuksiin ja synkronoituun tiedonkulkuun.

Jos Puppet-palvelun monitorointia halutaan lisätä, voidaan palveluun ottaa käyttöön sen tarjoama raportointiominaisuus. Tämän avulla Puppet-asiakkaat voidaan konfiguroida automaattisesti lähettämään raportteja haluttuun kansioon tai syslog-lokitiedostoon Puppet-palvelimelle. Raportteja voidaan halutessa myös määrätä lähetettäväksi haluttuihin sähköpostiosoitteihin. Raportointi tehostaa Puppet-palvelun monitorointia ja jos esimerkiksi jokin ei toimi, niin syytä tälle on huomattavasti helpompi lähteä selvittämään raportteja hyväksi käyttäen. [6, s. 121-122.]

5 Yhteenveto ja arvio

Linux-järjestelmien keskitetyn hallinnan verkko saatiin onnistuneesti toimimaan ja luvun 4.1 mainitussa listassa olevat tarkemmat tavoitteet saatiin täytettyä. Käytännön toteutusta varten tehdyssä aikataulussa pysyttiin suunnitelmien mukaisesti.

Käytännön toteutuksessa ongelmilta ei kuitenkaan vältytty. Muutaman kerran jouduttiin tilanteisiin, joissa ongelmien selvittäminen sai työn seisahtumaan paikoilleen. Ongelmatilanteet eivät kuitenkaan muodostuneet ylitsepääsemättömiksi ja ratkaisut niihin löydettiin kuitenkin suhteellisen nopeaan. Aikataulutuksessa kuitenkin huomioitiin ongelmatilanteiden mahdollinen esiintyminen ja niiden selvittämiseen tarvittava aika, jonka ansiosta aikataulussa pysyttiin suunnitelmien mukaisesti.

Työssä muodostetun ympäristön testauksessa ja ylläpidossa huomattiin, kuinka järjestelmien keskitetty hallinta on erittäin tehokas ja järjestelmällinen tapa hallita suuria

kokonaisuuksia pienellä vaivalla. Keskitetyn hallinnan idea, jossa järjestelmiä hallitaan yhdellä järjestelmällä monen sijasta, toteutui täysin. Myös järjestelmiin, varsinkin työasemiin kohdistuva töiden automatisointi tehosti ympäristön ja sen työasemien hallintaa.

Vaikka työssä käytetyt järjestelmät eivät olleet erillisiä fyysisiä laitteita vaan virtuaalisia järjestelmiä todettiin kuitenkin, että kyseinen keskitetyn hallinnan verkko on työn pohjalta täysin mahdollista toteuttaa onnistuneesti myös oikeilla fyysisillä laitteilla.

Keskitetyn hallinnan ympäristön muodostaminen työssä käytetyillä tuotteilla ja palveluilla on erittäin toimiva ja ilmainen ratkaisu, jos yrityksessä käytetään Unix-pohjaisia järjestelmiä. Ympäristössä on kuitenkin omat vahvuutensa ja heikkoutensa.

Ympäristön päällimmäisenä heikkoutena voidaan pitää NIS-palvelua. NIS-palvelun heikkouksia ovat sen nykypäivänä vaaditun tietoturvan vajavaisuus, käyttäjäinformaation vähäisyys sekä kehityksen päättyminen. Vaikka NIS-palvelua käytettiin tässä työssä ja käytetään vielä monissa Unix-pohjaisissa ympäristöissä, on sitä monessa tapauksessa alettu korvata LDAP-pohjaisilla ratkaisuilla. Voidaan siis sanoa, että LDAP ja muut tulevaisuuden hakemistopalveluprotokollat tulevat korvaamaan NIS:n tulevaisuudessa täysin. [5, s. 26, 29-30]

NIS-palvelun heikkouden johdosta sekä tulevaisuutta ajatellen, ympäristö olisi kannattavampi toteuttaa korvaamalla NIS-palvelu LDAP-pohjaisella ratkaisulla. Kyseisestä palvelusta on nimittäin olemassa Linux-pohjainen avoimen lähdekoodin toteutus OpenLDAP. [20; 25.]

Ympäristön suurimpana vahvuutena oli halutulla tavalla moiteetta toiminut Puppet, jonka tulevaisuus näyttää erittäin valoisalta. Se valittiin vuoden 2011 parhaaksi asetustiedostojen hallinnan palveluksi LinuxQuestions.org-verkkosivuston äänestyksessä. [32.] Toiseksi, Puppet-palvelun kehittäjä Puppet Labs pyrkii kokoajan kehittämään kyseistä palvelua ja tuomaan tulevaisuudessa esille uusia monipuolisia ominaisuuksia. Lisäksi, Puppet Labs tekee yhteistyötä yrityskumppaneidensa kanssa kuten Ciscon, VMwaren ja Google Venturesin. [33.] Jää siis nähtäväksi, kuinka yhteistyökumppanuudet vaikuttavat tuleviin Puppet-julkaisuihin ja yleisesti, mitä kaikkea tulevilla julkaisuilla voidaan tehdä.

Vaikka työssä muodostettu ympäristö on toimiva keino rakentaa täysin ilmainen keskitetyn hallinnan ympäristö, on organisaatiolle sellaisen rakentamisessa kuitenkin hyvä pitää mielessä työn alussa mainittu Active Directory. LDAP-pohjaisuutensa ansiosta sekä Microsoftin monopoliaseman myötä Active Directory alkaa olla tänä päivänä yleisin hakemisto- ja nimipalvelu erinäisissä verkoissa. Tähän yhtenä syynä on se, että Active Directory voidaan käyttöönottaa monilla järjestelmillä, Windows- sekä Unix-pohjaisilla. Eli, hyvin yleisen mallin mukaan, jossa verkossa on sekä Windows- että Unix-pohjaisia järjestelmiä, Active Directory hoitaa pääosin näiden kaikkien keskitetyn hallinnan. On siis hyvä pitää mielessä, että vaikka Active Directory ei palveluna ole täysin ilmainen, sen mukana tuomat ominaisuudet ja hyödyt saattavat olla organisaatiolle hintansa arvoisia. [27; 26; 29.]

Lähteet

1 Campi Nate, Bauer Kirk. 2009. Automating Linux and Unix System Administration Second edition, Apress.

2 Lattu, Matti. 2010. Tietotekniikkaosasto - Keskitetysti hallitun työaseman anatomia. Verkkodokumentti. <http://www.helsinki.fi/atk/lehdet/110/artikkeli1.html>. 21.4.2010. Luettu 20.2.2012.

3 Oukka, Asko. 2010. Palvelinympäristöjen keskitetty kokoonpanon hallinta avoimen lähdekoodin ohjelmistoilla. Verkkodokumentti. <https://publications.theseus.fi/bitstream/handle/10024/16635/Oukka_Asko.pdf>. 11.6.2010. Luettu 25.2.2012.

4 Microsoft Solution for Windows-based Hosting v. 4.0. 2006. Verkkodokumentti. Microsoft. <download.microsoft.com/download/5/c/4/5c49d9c8-a3e9-4806-8c6d-a621b8d2402b/Windows-basedHosting_CentralizedManagement.doc>. Luettu 27.2.2012.

5 Stern Hal, Eisler Mike, Ricardo Labiaga. 2001. Managing NFS and NIS 2nd Edition, O'Reilly Media.

6 James Turnbull. 2007. Pulling Strings with Puppet, Apress.

7 Compare Puppet Enterprise vs. Puppet Open Source. 2012. Verkkodokumentti. Puppet Labs. <http://puppetlabs.com/puppet/enterprise-vs-open-source>. Luettu 24.2.2012.

8 Raynal, Frédéric. 2001. System Administration: Yellow Pages part 1. Verkkodokumentti. <http://www.ibiblio.org/pub/Linux/docs/LDP/linuxfocus/English/July2001/article148.shtml>. 29.6.2001. Luettu 22.2.2012.

9 The Network Information System. Verkkodokumentti. Faqs.org. <http://www.faqs.org/docs/linux_network/x-087-2-nis.html>. Luettu 23.2.2012.

10 Suess, Jack. 1995. Using NIS. Verkkodokumentti. <http://userpages.umbc.edu/~jack/ifsm498d/llb-nis.html>. 25.9.1995. Luettu 23.2.2012.

11 Administering the Network Information Service (NIS). 2004. Verkkodokumentti. The SCO Group. <http://uw714doc.sco.com/en/NET_nis/CONTENTS.html>. Luettu 24.2.2012.

12 NFS Administration Guide. 1994. Verkkodokumentti. Sun Microsystems. <http://docs.oracle.com/cd/E19457-01/801-6634/801-6634.pdf>. Luettu 22.2.2012.

13 NFS: Network File System. Verkkodokumentti. Javvin Company. <http://www.javvin.com/protocolNFS.html>. Luettu 24.2.2012.

14 Smith, Christopher. Linux NFS faq. Verkkodokumentti. <http://nfs.sourceforge.net>. Luettu 24.2.2012.

15 Leskinen, Jyri. NFS (Network File System). Verkkodokumentti. <http://users.jyu.fi/~jyril/opiskelu/NFS.html>. Luettu 24.2.2012.

16 Ratilainen, Tero. 2010. Palvelinklusterin rakentaminen Debian-ympäristöön. Verkkodokumentti. <https://publications.theseus.fi/bitstream/handle/10024/23999/Ratilainen_Tero.pdf>. 15.11.2010. Luettu 24.2.2012.

17 GlusterFS General FAQ - GlusterDocumentation. 2011. Verkkodokumentti. Gluster Community. <http://www.gluster.org/community/documentation/index.php/GlusterFS_General_FAQ>. Luettu 24.2.2012.

18 Frequently Asked Questions - Documentation - Puppet Labs. 2011. Verkkodokumentti. Puppet Labs. <http://docs.puppetlabs.com/guides/faq.html>. Luettu 25.2.2012.

19 Chapter 4 - A Directory Sevice. 2011. Verkkodokumentti. OpenSuse.org. <http://doc.opensuse.org/documentation/html/openSUSE/opensuse-security/cha.security.ldap.html#sec.security.ldap.vs_nis>. Luettu 27.2.2012.

20 Heiss, Jason. 2004. Replacing NIS with Kerberos and LDAP HOWTO. Verkkodokumentti. <http://aput.net/~jheiss/krbldap/howto.html>. 11.6.2004. Luettu 27.2.2012.

21 SettingUpNISHowTo - Community Ubuntu Documentation. 2011. Verkkodokumentti. Ubuntu Community. <https://help.ubuntu.com/community/SettingUpNISHowTo>. Luettu 16.1.2012.

22 SettingUpNFSHowTo - Community Ubuntu Documentation. 2012. Verkkodokumentti. Ubuntu Community. <https://help.ubuntu.com/community/SettingUpNFSHowTo>. Luettu 20.1.2012.

23 Pöntinen, Ville. 2006. Ubuntu-lähiverkko (NIS + NFS, käyttäjien luontia). Verkkodokumentti. <http://forum.ubuntu-fi.org/index.php?topic=4867.0>. 26.8.2006. Luettu 16.1.2012.

24 Puppet. 2010. Verkkodokumentti. Ubuntu Documentation Team. <https://help.ubuntu.com/10.10/serverguide/C/puppet.html>. Luettu 26.1.2012.

25 OpenLDAP Software 2.4 Administrator's Guide. 2012. Verkkodokumentti. The OpenLDAP Project. <http://www.openldap.org/doc/admin24/index.html>. Luettu 4.3.2012.

26 Tolvanen, Perttu. 2011. Käsitteet ojennukseen: Active Directory (AD), LDAP, SSO ja identiteetinhallinta | Viides taso. Verkkodokumentti. <http://viidestaso.wordpress.com/2011/04/29/kasitteet-ojennukseen-active-directory-ad-ldap-sso-ja-identiteetinhallinta>. 21.4.2011. Luettu 4.3.2012.

27 Active Directory Benefits for Smaller Enterprises. 2004. Verkkodokumentti. Microsoft. <http://download.microsoft.com/download/9/c/c/9cc119b8-2c03-4f12-b4dd-14ba60c536bc/AD%20Business%20Benefits%20for%20Mid-market.doc>. Luettu 5.3.2012.

28 Using Snapshots. 2012. Verkkodokumentti. VMware Inc. <http://www.vmware.com/support/ws55/doc/ws_preserve_using_sshot.html>. Luettu 6.3.2012.

29 Salonen, Janne. 2012. Yliopettaja, Metropolia AMK, Helsinki. Keskustelu 5.3.2012.

30 Kukuk, Thorsten. 2012. www.linux-nis.org (Linux NIS+ Support). Verkkodokumentti. <http://www.linux-nis.org/nisplus/index.html>. 22.1.2012. Luettu 3.4.2012.

31 NIS versus NIS+. Verkkodokumentti. Faqs.org. <http://www.faqs.org/docs/linux_network/x-087-2-nis.nisplus.html>. Luettu 3.4.2012.

32 2011 LinuxQuestions.org Members Choice Award Winners. Verkkodokumentti. LinuxQuestions.org. <http://www.linuxquestions.org/questions/linux-news-59/2011-linuxquestions-org-members-choice-award-winners-928502>. 2.9.2012. Luettu 3.4.2012.

33 Endsley, Rikki. 2012. A Peek Behind the Curtain at Puppet Labs. Verkkodokumentti. <https://www.linux.com/news/enterprise/systems-management/555803-a-peak-behind-the-curtain-at-puppet-labs>. 19.3.2012. Luettu 3.4.2012.

34 <http://images.gittigidiyor.com/3489/DELL-PowerEdge-2950-Server__34895538_0.jpg>

35 <http://www.gettyicons.com/free-icons/133/shimmer/png/256/folder_256.png>

NIS-palvelun asennus ja konfigurointi [21]

NIS-palvelun asennus ja konfigurointi palvelimelle ja asiakkaalle. Näissä ohjeissa NIS-palvelun domain-nimi on sama kuin itse palvelimelle annettu nimi, *firstoffice*.

Palvelin:

Asennus komennolla, jonka aikana määritellään domain-nimi:

```
sudo apt-get install nis
```

Palvelimen oman NIS-nimensä tunnistus rivin lisäyksellä /etc/hosts-tiedostoon:

```
127.0.0.1 localhost.localdomain     localhost   firstoffice
```

Palvelimen itsensä tunnistus NIS-palvelimeksi rivin lisäyksellä /etc/yp.conf-tiedostoon:

```
ypserver firstoffice server 10.95.202.253
```

Käyttäjän sidonta oikeisiin ryhmiin rivin muokkauksella /var/yp/makefile-tiedostoon:

```
MINGID=1
```

Tietoturvan parannus rivin lisäyksellä /etc/ypserv.securenets-tiedostoon:

```
255.255.255.0           10.95.202.0
```

NIS-palvelun uudelleen käynnistys komennolla:

```
sudo service nis restart
```

NIS-karttojen luonti ensimmäistä kertaa komennolla:

```
sudo /usr/lib/yp/ypinit –m
```

NIS-karttojen päivitys komennolla, kun käyttäjä- ja/tai ryhmätietoihin tehdään muutoksia:

```
sudo make -C /var/yp
```

Asiakas:

Asennus komennolla:

```
sudo apt-get install nis
```

Jotta NIS osaa etsiä tietoa oikein, /etc/nsswitch.conf-tiedostoon lisäys ja tarkistus, että alla mainitut tiedot ovat näin:

```
passwd:         compat
group:          compat  nis
shadow:         compat
netgroup:       nis
```

Domain-nimen asettaminen rivin lisäyksellä /etc/defaultdomain-tiedostoon:

```
firstoffice
```

NIS-palvelimen paikannus rivin lisäyksellä /etc/yp.conf-tiedostoon:

```
ypserver firstoffice
```

NIS-palvelimen nimeäminen rivin lisäyksellä /etc/hosts-tiedostoon:

```
10.95.202.253 firstoffice
```

Jotta NIS hakisi käyttäjätietoja myös NIS-palvelimelta, rivin lisäys /etc/passwd-tiedoston loppuun:

```
+::::::
```

Jotta NIS hakisi salasanatietoja myös NIS-palvelimelta, rivin lisäys /etc/shadow-tiedoston loppuun:

```
+::::::::
```

Jotta NIS hakisi ryhmätietoja NIS-palvelimelta, rivin lisäys /etc/group-tiedoston loppuun:

```
+:::
```

NIS ja Portmap palveluiden uudelleen käynnistys:

```
sudo service portmap restart
sudo service nis restart
```

NFS- ja Automount-palveluiden asennus ja konfigurointi [22]

NFS-palvelun asennus ja konfigurointi palvelimelle ja asiakkaalle.

Palvelin:

Asennus komennolla:

```
sudo apt-get install nfs-kernel-server
```

Hakemisto jakoon rivin lisäyksellä /etc/exports-tiedostoon:

```
/users/home     10.95.202.0/24(rw,sync,no_root_squash)
```

Osoitteiden ja verkkojen salliminen rivin lisäyksellä /etc/hosts.allow-tiedostoon:

```
portmap: 10.95.202.0/255.255.255.0
```

Osoitteiden ja verkkojen estäminen rivin lisäyksellä /etc/hosts.deny-tiedostoon:

```
portmap: ALL
```

Asiakas:

Asennus komennolla:

```
sudo apt-get install nfs-common
```

Osoitteiden ja verkkojen salliminen rivin lisäyksellä /etc/hosts.allow-tiedostoon:

```
portmap: 10.95.202.252
```

Osoitteiden ja verkkojen estäminen rivin lisäyksellä /etc/hosts.deny-tiedostoon:

```
portmap: ALL
```

Hakemiston luonti kotihakemistojen liitosta varten:

```
sudo mkdir /users
sudo mkdir /users/home
```

Hakemiston automaattinen liittäminen fstab-tiedoston avulla:

Rivin lisäys /etc/fstab-tiedostoon:

```
10.95.202.252:/users/home           /users/home             nfs
bg,rw,intr,exec,nosuid,rsize=8192,wsize=8192 0 0
```

Hakemiston automaattinen liittäminen Automount-palvelun avulla:

Palvelun asennus:

```
sudo apt-get install autofs5
```

Rivin lisäys auto.master-tiedostoon:

```
/users/home     /etc/auto.home  --timeout=5
```

Auto.home-tiedoston luonti ja rivin lisäys:

```
*           -fstype=nfs,rw,hard,user,noatime,exec,nodev,noauto,nosuid,async
10.95.202.252:/users/home/&
```

Puppet-palvelun asennus ja konfigurointi [6; 24]

Puppet-palvelun asennus ja konfigurointi palvelimelle ja asiakkaalle.

Palvelin:

Asennus komennolla:

```
sudo apt-get install puppetmaster
```

Palvelimen oman Puppet-nimensä tunnistus rivin lisäyksellä /etc/hosts-tiedostoon:

```
127.0.0.1 localhost.localdomain localhost puppet
```

Hakemiston jako asetustiedostoja varten rivien lisäyksellä /etc/puppet/fileserver.conf-tiedoston files-osion alle:

```
path /etc/puppet/files
allow *
```

Hakemistojen ja tiedostojen luonti komennoilla:

```
sudo mkdir /etc/puppet/files
sudo mkdir /etc/puppet/manifests
sudo touch /etc/puppet/manifests/site.pp
sudo mkdir /etc/puppet/manifests/classes
```

Asiakas:

Asennus komennolla:

```
sudo apt-get install puppet
```

Puppet-palvelimen nimeäminen rivin lisäyksellä /etc/hosts-tiedostoon:

```
10.95.202.253 puppet
```

Asiakkaan tunnistus ja suoritusintervallin määritys rivien lisäyksellä /etc/puppet/puppet.conf-tiedostoon:

```
certname=[järjestelmän host-nimi]
runinterval=1800
```

Koneen käynnistyessä palvelun käynnistymisen automatisointi rivin muokkauksella /etc/default/puppet-tiedostoon:

```
START=YES
```

Palvelun uudelleen käynnistäminen uusilla asetuksilla komennolla:

```
sudo service puppet restart
```

Sertifikaattipyyntö Puppet-palvelimelta komennolla:

```
sudo puppetd --test
```

Avoimien sertifikaattipyyntöjen tarkistus ja niiden hyväksyntä Puppet-palvelimen puolella komennoilla:

```
sudo puppetca --list
sudo puppetca --sign [listassa näkyvän järjestelmän nimi]
```

Puppet-palvelun toimivuuden testaus asiakkaan puolella komennolla:

```
sudo puppetd --test
```

Puppet-palvelimelle luokkien luonti:

Luodaan luokat niiden ylläpitotoimenpiteiden mukaan ja sijoitetaan ne hakemistoon /etc/puppet/manifests/classes. Muokataan site.pp-tiedosto siten, että jokaisella järjestelmällä on oma osionsa ja osioihin määritellään halutut luokat. Lopullisen site.pp-tiedoston sisältö seuraavalla sivulla.

--------- site.pp alkaa ---------

```
import "classes/*"

node 'test-workstation-desktop' {
}
node 'workstation2-desktop' {
        include packages
        include directories
        include files
        include scripts
        include services
        include update
}
node 'workstation3-desktop' {
        include packages
        include directories
        include files
        include scripts
        include services
        include update
}
node 'nfs' {
        include nfs_files
        include update
}
```

--------- site.pp loppuu ----------

Automatisointiskriptit

Työaseman konfigurointiin ja käyttöönottoon hyödynnettävät skriptit

Puppet-palvelimelta asiakkaalle tarjottava yp_add.sh Bash-skripti, joka suoritetaan sen kopioiduttua asiakkaan järjestelmään. Skripti lisää automaattisesti asiakkaan NIS-palveluun liittyvät asetusrivit oikeisiin tiedostoihin.

--------- yp_add.sh alkaa ----------

```
#!/bin/bash

lastline1=`sudo tail -1 /etc/shadow`
lastline2=`sudo tail -1 /etc/group`
lastline3=`sudo tail -1 /etc/passwd`
lastline4=`sudo tail -1 /etc/hosts`

if [ "$lastline1" != "+:::::::::" ]
then
        sudo echo +::::::::: | cat >> /etc/shadow
fi

if [ "$lastline2" != "+:::" ]
then
        sudo echo +::: | cat >> /etc/group
fi

if [ "$lastline3" != "+::::::" ]
then
        sudo echo +:::::: | cat >> /etc/passwd
fi

if [ "$lastline4" != "10.95.202.253 firstoffice" ]
then
        sudo echo 10.95.202.253 firstoffice | cat >> /etc/hosts
fi
```

--------- yp_add.sh loppuu ----------

Työaseman puhtaan käyttöjärjestelmäasennuksen jälkeen suoritettava automation.sh Bash-skripti, jonka avulla verkkoasetuksien asettaminen, järjestelmän päivittäminen, Puppet-palvelun asennus sekä Puppet-palvelun konfigurointi tapahtuvat automaattisesti.

--------- automation.sh alkaa ----------

```
#!/bin/bash

echo "Verkko on 10.95.202.0. Anna IP-osoitteen viimeinen oktetti:"
read ipadd

echo "\nauto eth0\niface eth0 inet static\naddress 10.95.202.$ipadd\nnetmask
255.255.255.0\nnetwork 10.95.202.0\nbroadcast 10.95.202.255\ngateway
10.95.202.254" | cat >> /etc/network/interfaces

echo "nameserver 10.95.254.252\nnameserver 10.95.254.253" | cat >>
/etc/resolv.conf

apt-get update
apt-get upgrade -y
apt-get install puppet

HOSTN=`hostname`
sed -i 's/START=no/START=yes/g' /etc/default/puppet
echo "certname=$HOSTN\nruninterval=1800" | cat >> /etc/puppet/puppet.conf
echo "10.95.202.253 puppet"| cat >> /etc/hosts

echo "Skripti suoritettu onnistuneesti! Kone bootataan 5 sekunnin kuluttua.."
sleep 5
reboot
```

--------- automation.sh loppuu ----------