Eero Volotinen

# Kotikartanon tietojärjestelmän uudistus

Metropolia Ammattikorkeakoulu
Insinööri (AMK)
Tietotekniikan koulutusohjelma
Insinöörityö
3.11.2012

Metropolia

**Tiivistelmä**

| | |
|---|---|
| Tekijä<br>Otsikko<br><br>Sivumäärä<br>Aika | Eero Volotinen<br>Kotikartanon tietojärjestelmän uudistus<br><br>20 sivua + 5 liitettä<br>3.11.2012 |
| Tutkinto | Insinööri (AMK) |
| Koulutusohjelma | Tietotekniikka |
| Suuntautumisvaihtoehto | |
| Ohjaaja | Koulutuspäällikkö Markku Karhu |


Työn aiheena on Kotikartanon tietojärjestelmän uudistus. Tarkoituksena oli suunnitella uusi järjestelmä ja korvata vanha järjestelmä. Uudistukseen liittyvän vanhan verkkoyhteyden poistaminen sekä palomuurien, aktiivilaitteiden ja palvelinohjelmistojen päivittäminen vastaamaan nykypäivän tasoa.

Kotikartano-yhdistys on yleishyödyllinen yhdistys, jonka budjetit ovat pieniä, siksi tarkoituksena on toteuttaa kustannustehokas ja toimiva järjestelmä. Järjestelmän arvioitava käyttöikä ilman laitteistopäivityksiä on noin viisi vuotta.

Uudistuksia varten vanhan järjestelmän ongelmakohdat analysointiin huolellisesti, ja ne otettiin huomioon uutta toteutusta rakentaessa.

Toteutuksena käytettiin Linux-järjestelmää ja avoimen lähdekoodin tuotteita. Tärkeimpiä osia toteutuksesta olivat Gentoo-Linux ja LTSP-projektin tuottamat ohjelmistot, jotka mahdollistavat vanhojen koneiden uusiokäytön.

Uudistuksen toteuttamisen jälkeen saatiin käyttöön toimiva verkkoympäristö ja aiempaa nopeammat palvelut edulliseen hintaan. Hinta oli vain murto-osa, jos sitä verrataan kilpailevaan Microsoft Windows -ympäristöön ja sen lisensointikustannuksiin.

Uudistettu toteutus onnistui hyvin, ja Kotikartano-yhdistys sai käyttöönsä edullisen ja toimivan järjestelmän, jonka käyttöikä on pitkä ja kokonaiskustannukset matalat.



| | |
|---|---|
| Avainsanat | Gentoo, Linux, X-päätteet, LTSP, Verkko |

Metropolia

**Abstract**

| | |
|---|---|
| Author(s)<br>Title<br><br>Number of Pages<br>Date | Eero Volotinen<br>Improvement of Kotikartano network<br><br>20 pages + 5 appendices<br>3 November 2012 |
| Degree | Bachelor of Engineering |
| Degree Programme | Information technology |
| Specialisation option | |
| Instructor(s) | Markku Karhu, Head of Degree Programme in Information Technology |


The subject of this thesis was redesigning and implementing a network neighbourhood for the Kotikartano association. The purpose was to design a new system and replace the old one. In addition, the old network connection was removed, and firewalls, servers and wireless network devices were updated to correspond to the modern needs.

Kotikartano is a charity-based organization with a low budget. Thus, the goal was to implement a cost-effective and functional system. The estimated working life of the new system is five years without hardware updates.

After setting up the new system, a functional network neighbourhood and higher-speed services were introduced at a fair price. This price was only a fraction of the costs compared with the competing Microsoft Windows neighbourhood and the related licensing costs.

The Linux system and open source code products were utilised. The most important parts of the system were Gentoo-Linux and the software produced by the LTSP project, which enabled reusing of old computers.

The old system and the related problems were analyzed with care and were taken into consideration in building up the new system.

The redesigning project was successful and the Kotikartano association got an advantageous and functional system with long working life at low total costs.



| | |
|---|---|
| Keywords | Gentoo, Linux, X-terminals, LTSP |

## Sisällys

## Lyhenteet

| | |
|---|---|
| ADSL | Asymmetric Digital Subscriber Line, verkkotekniikka joka käyttää puhelinkuparia nettiyhteyteen. |
| AMD | Advanced Micro Devices, prosessorien valmistaja. |
| BOOTP | Bootstrap Protocol, käynnistysprotokolla. |
| DHCP | Dynamic Host Configuration Protocol, dynaaminen verkko-osoitteiden määritys. |
| LTSP | Linux Terminal Server Project, projekti joka keskittyy Linux-päätelaitteisiin. |
| MSCE | Microsoft Certified Systems Engineer, Microsoftin valtuuttama järjestelmäasiantuntija. |
| NAT | Network Address Translation, verkko-osoitteiden muuntotapa. |
| NFS | Network FileSystem, verkotettu tiedostojärjestelmä. |
| PDC | Primary Domain Controller, toimialueen kirjautumis- ja hallintapalvelin. |
| PROXY | Välimuistipalvelin. |
| PXE | Preboot eXecution Environment, verkosta käynnistymisen mahdollistava verkkolataaja. |
| SAMBA | Ohjelmisto, joka toteuttaa verkkojaot Windows-yhteensopivasti. |
| SPI | Stateful Packet Inspection, tilatietoinen pakettisuodatus. |
| TFTP | Trivial File Transfer Protocol, yksinkertainen tiedostojensiirtokäytäntö. |

VLAN Virtual Local Area Network, lähiverkko, jossa kytkimen portit voidaan virtuaalisesti erottaa toisistaan.

WLAN Wireless Local Area Network, langaton lähiverkko.

# 1 Johdanto

Työn toimeksiantaja oli Joensuussa sijaitseva Kotikartanoyhdistys, joka tarjoaa työtoimintaa, kuntoutusta ja koulutusta pitkäaikaistyöttömille. Tietotekniikan käyttäjäkunta on eri-ikäistä ja osaamistasoltaan vaihtelevaa. Osa käyttäjistä hallitsee tietokoneen perusasiat hyvin, ja jotkut eivät ole koskaan tietokonetta edes käyttäneet, vaan heidät on vasta tarkoitus kouluttaa tulemaan tietokoneen kanssa toimeen ainakin perusasioissa. Myös käyttäjien kielitaito on vaihtelevan tasoinen. Nämä edellä mainitut seikat on otettava huomioon järjestelmän uudistamisessa.

Työn tarkoituksena on rakentaa, suunnitella ja toteuttaa kustannustehokas verkkoympäristö, jolla voitaisiin korvata Kotikartanoyhdistyksen vanhentunut järjestelmä. Uusi toteutus oli tarkoitus tehdä mahdollisimman pienellä budjetilla, mutta silti laadusta tinkimättä. Laitteistona käytettiin keskihintaisia ja tunnettujen valmistajien tuotteita.

Työn suuruutta kuvaa hyvin se, että järjestelmä rakennettiin täysin puhtaalta pöydältä aloittaen suunnitteluvaiheesta ja kilpailuttaen Internet-operaattorit. Tarkoituksena oli luopua vanhasta verkkoyhteydestä ja samalla korvata verkon aktiivilaitteet uudemmilla.

Kotikartanoyhdistyksellä on noin 20 vapaasti käytössä olevaa tietokonetta, ja lisäksi toimihenkilöillä on 12 henkilökohtaista ja yksi kannettava tietokone.

Toteutukseksi valittiin Linux-ympäristö siksi, että vastaava kaupallinen toteutus olisi tullut niin kalliiksi, ettei siihen olisi voinut edes alennetuilla hinnoilla ryhtyä ilman ulkopuolista rahoitusta. Linux on todettu suuremmissakin järjestelmissä hyväksi ja luotettavaksi ympäristöksi. Hieman kokeellisuutta lisättiin työhön valitsemalla palvelinratkaisun alustaksi 64-bittinen AMD-prosessorialusta. Suunnitelmissa oli käyttää ympäristöä myös graafisesti. Alun perin ei ollut kokemusta toimisiko järjestelmä siinä tarkoituksessa riittävän hyvin.

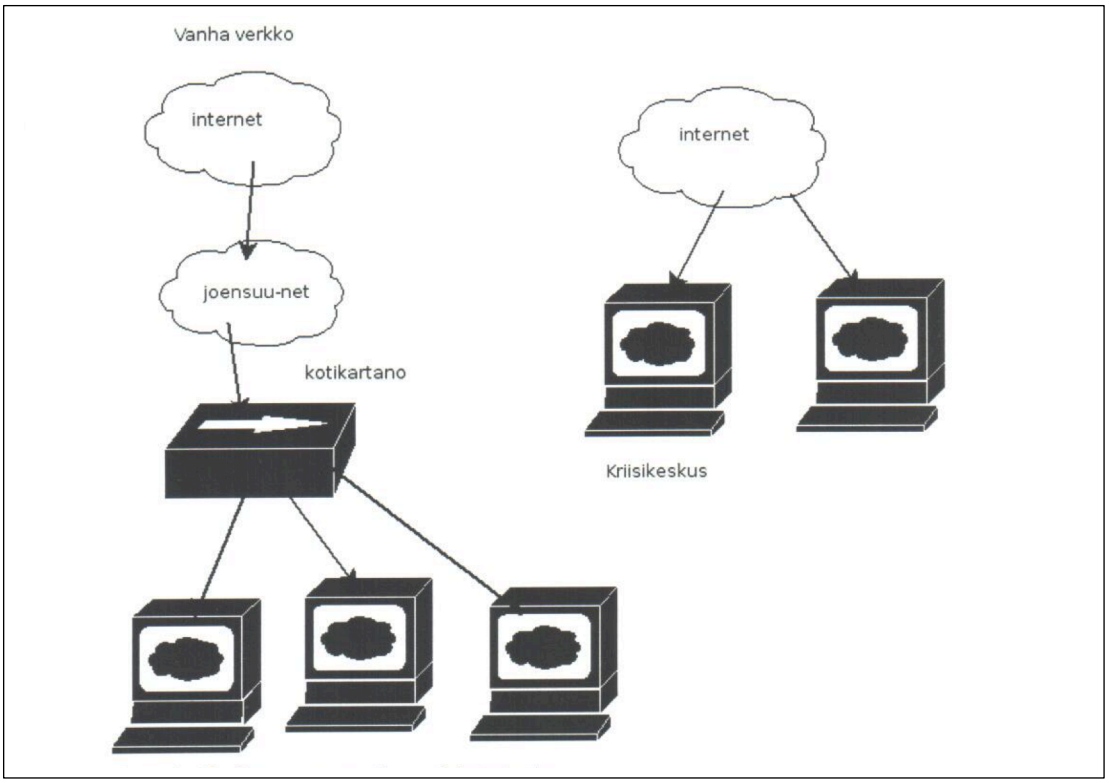
Toteutuksessa oli lisäksi vaatimuksena, että vanha ratkaisu pysyy käytettävänä rinnalla niin pitkään kuin se vain on teknisesti mahdollista. Käytössä olevaa järjestel-

mää ei voida heti korvata uudella, ja lisäksi pidettiin varalla sitä mahdollisuutta, että jos uusi toteutus ei ole riittävän hyvä, niin on mahdollista vielä palata vanhaan järjestelmään ilman suuria muutostöitä.

Projektin kestoksi oli arvioitu noin kuusi kuukautta. Suuri osa isoimmista töistä tehtiin viikonloppuisin ja iltaisin, etteivät ne olisi häirinneet vanhan järjestelmän toimintaa.

# 2 Vanha verkko

Vanha verkko oli rakennettu vuosina 2001 - 2002, ja se oli monilta osin jo vanhentunutta tekniikkaa. Lisäksi oli varsin helposti nähtävissä, että suunnittelussa ja toteutuksessa oli käytetty erittäin vaihtelevaa osaamista. Kuviossa 1 on vanhan verkon rakenne, jossa Kotikartanoyhdistys ja Kriisikeskus ovat vielä täysin omissa verkoissaan.




Kuvio 1. Vanha verkko.

Vanhassa verkossa oli Linux-pohjaisia tiedostopalvelimia ja sisäverkon Intranet-palvelin. Verkkoyhteydet tulivat Joensuun kaupungin kautta eikä niitä ollut erotettu toisistaan palomuurilaitteilla. Tämän ratkaisun ongelmana oli keskinäinen liikenne. Verkossa liikkui viruksia ja haittaliikennettä ja niiden rajoittamiseksi ei käytännössä pystytty tekemään mitään. Toinen ongelma oli se, että Kotikartanoon kuuluu kaksi rakennusta, joiden verkko-yhteydet olivat erillisiä ratkaisuja. Rakennusten väliset yhteydet eivät toimineet kuin sähköpostin tai puhelimen välityksellä, ja verkkoyhteyksistä joutui maksamaan kaksinkertaiset kustannukset.

Toimihenkilöiden työasemissa oli Windows NT 4.0 -käyttöjärjestelmä, joka oli jo elinkaarensa päässä. Tietoturvapäivityksiä ei enää ollut saatavissa, ja laitekanta olisi tarvinnut uusimista, jos nykyistä käyttöjärjestelmäperhettä olisi ruvettu päivittämään. Myös varusohjelmistot olivat elinkaarensa päässä.

Hyvä asia verkossa oli se, että ihmiset olivat jo tottuneet OpenSourceen [1] eli avoimen lähdekoodin tuotteisiin, kuten esimerkiksi Firefoxiin [2] ja OpenOfficeen [5].

Nykyinen ympäristö käytiin läpi tutkien sen ongelmakohdat, jotta niitä työn edetessä voitaisiin parantaa ja budjetin sallimissa rajoissa.

# 3 Työn vaiheet

Ensimmäinen tehtävä oli kilpailuttaa verkkoyhteydet ja hankkia tarvittava laitteisto, että verkkoyhteydet voitaisiin vaihtaa eri operaattorin verkkoon ja tehdä järjestelmästä operaattoreista riippumaton. Tarjoukset internet-yhteyksistä pyydettiin Elisalta, Soneralta ja Telekarelialta. Tarjouskilpailun voitti Elisa, joka tarjosi koti-ADSL liittymää. Hintaero nykyiseen verkkoyhteyteen verrattuna oli noin puolet nykyisestä hinnasta. Internetyhteyden kapasiteetti tosin oli noin puolta pienempi. Budjettisyistä jouduttiin tyytymään tällaiseen nettiyhteyteen.

Seuraavaksi tilattiin aktiivilaitteet edullisimmasta paikasta. Palomuuriksi valittiin Zy-

xelin Zywall 2 -palomuurilaite, kytkimeksi 3comin hallittava 24-porttinen 10/100/1000-kytkin, jossa oli kaksi gigaista porttia. ADSL-päätelaitteeksi valittiin Telewell EA-200.

Internet-yhteydet asennettiin paikalleen työajan ulkopuolella, ja tämä oli vaihe, jossa oli mahdollisuus kokea ongelmia, sillä vanha Internet-yhteys ei olisi tämän muutostyön jälkeen enää ollut käytettävissä. Uusi verkkoyhteys tuli sillattuna ADSL-päätelaitteeseen, johon kytkettiin palomuuri. Palomuurista kytketään verkkoyhteys kytkimeen, josta tietokoneet saavat Internet-yhteytensä. Muutos aiheutti melko paljon konfiguraatiomuutoksia kaikkiin tietokoneisiin ja myös vanhoihin palvelimiin.

Seuraavaksi MMP-systems kytki WLAN-sillan Kriisikeskuksen ja Kotikartanon yhdistäen kaksi taloa samaan lähiverkkoon. Näin pystyttiin samalla luopumaan toisesta Internet-yhteydestä ja pudotettiin säännöllisesti toistuvaa kustannusrakennetta.

Näiden kahden ison muutostyön jälkeen oli usean kuukauden vaihe, jossa testattiin uuden verkon toimintaa eikä vielä tehty muita muutoksia. Testausvaiheen päätyttyä alettiin suunnitella uusia muutoksia, ja samalla ostettiin kaksi WLAN-tukiasemaa, joilla verkotettiin Kotikartano myös langattomasti. Muutoksen tarkoituksena oli tuoda joustavuutta nettiyhteyksiin.

Seuraava vaihe oli verkkopalvelinten uusiminen. Tätä tarkoitusta varten ostettiin 64-bittinen tietokone, josta oli tarkoitus tehdä verkon palvelin. Tämä vaihe oli kaikkein työläin. Laitteisto asennettiin ja vanhat palvelut siirrettiin uuteen laitteeseen. Työvaiheen päättäminen edellytti kolmen kuukauden testijaksoa.

Viimeisessä vaiheessa käyttäjät pikkuhiljaa siirrettiin uuden verkon palvelujen piiriin ja vanhan verkon palvelimet purettiin pois käytöstä. Vanhojen Intranet-palveluiden siirtäminen uuteen palvelinjärjestelmään vei kuukauden, koska niiden toteutus oli vaihtelevaa ohjelmistojen laadun suhteen. Suurimpaan osaan verkkopalveluista piti tehdä ohjelmistomuutoksia, koska muuten niiden saaminen toimimaan uudessa järjestelmässä olisi ollut täysin mahdotonta. Samalla poistettiin ohjelmista verkko-osoiteriippuvuutta ja tehtiin muita korjauksia.

Projektin loppuvaiheessa käytettiin viikon verran aikaa palvelinohjelmistojen hienoviritykseen, jotta niistä saataisiin enemmän tehoa irti niin palvelu- kuin verkkoyhteyksiin.

# 4 Tekninen toteutus

Verkon ensimmäinen osa oli Telewell EA-200 ADSL -modeemi. Se asennettiin toimimaan siltatilassa, jolloin verkko-osoite tulee laitteen läpi ilman muutoksia. Tähän päädyttiin siksi, että laitteen heikko prosessoriteho oli tiedossa, eikä se pystyisi suorittamaan NAT-osoitteenmuunnosta [6] verkon koneille.

Seuraava verkon osa oli Zyxel Zywall 2 spi -palomuuri, joka saa verkko-osoitteensa ADSL-modeemin läpi. Palomuuri määriteltiin tekemään NAT-osoitteenmuunnos niin, etta se peittää koneiden osoitteen ja sallii vain sisäverkosta avatut yhteydet ulkomaailmaan. Tämä estää suurimman osan verkkoyhteyksistä ja estää myös käytännössä suorat yhteydet ulkomaailmasta. Toiminnan tarkoituksena on suojata työasemia ja palvelimia Internetin tuomilta vaaroilta, kuten esimerkiksi Blaster-madolta [8], joka osaa hyökätä suojaamattomiin Windows-järjestelmiin ilman käyttäjän aktiivisuutta.

Palomuurista yhteydet tulevat 3com VLAN -kytkimeen, johon ei tehty asetusmuutoksia vaan päätettiin jakaa verkko tasaisesti käyttäjien kesken. Tulevaisuudessa on myös mahdollista ruveta käyttämään VLAN-ominaisuutta, mikäli tarve sen vaatii. Tässä vaiheessa vanhaan verkkoon täytyi tehdä muutoksia ja muuttaa kaikki vanhat koneet käyttämään DHCP-protokollaa. Myös palvelinten verkko-osoitteet täytyi muuttaa uuteen verkkoon sopiviksi. Kytkimestä yhteydet menevät kaikille tietokoneille ja lisäksi WLAN-tukiasemille, kahdelle kirjoittimelle ja WLAN-sillalle.

Kriisikeskus ja Kotikartano yhdistettiin WLAN-sillalla, jonka toteuttamiseen käytettiin Buffalon WLAN-tukiasemia ja ulkoisia antenneja. Matkaa talojen välillä on noin 145 metriä, ja toiminta saatiin suunta-antenneilla vakaaksi ja nopeaksi. Kriisikeskukseen ei asennettu muita aktiivilaitteita, vaan siellä verkko jaetaan muutamalle koneelle suoraan Buffalo-tukiaseman sisäisestä kytkimestä. Siltatoiminnolla käytännössä kaikki verkon

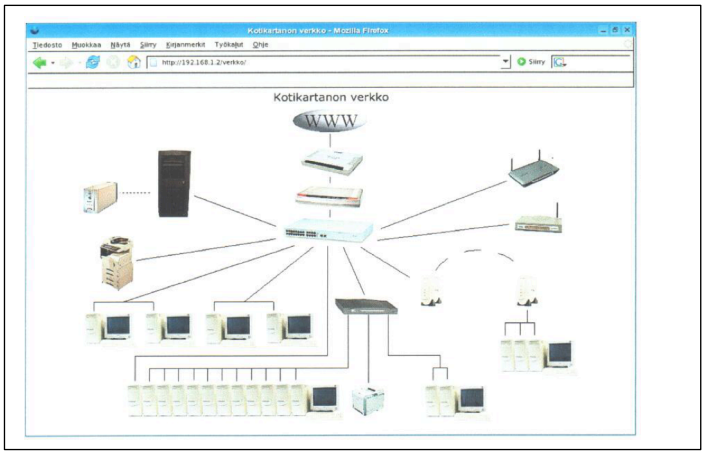
toiminnot voidaan tarjota myös Kriisikeskukseen ilman kiinteitä johtoja, joiden asennuskustannukset olisivat suuret.

Palvelimia asennettiin palvelinhuoneeseen kaksi kappaletta ja säilytettiin vielä vanhat palvelimet, jotka on tarkoitus poistaa, kun kaikki tarvittava saadaan toimimaan riittävällä tasolla. Palvelin 1, lempinimeltään Uguru, on verkon pääpalvelin, joka tarjoaa verkolle seuraavat palvelut:

- tiedostojen jako
- kirjautuminen verkkoon
- käyttäjäoikeuksien rajaaminen
- WWW-välimuisti
- mainoksien suodatus
- sähköpostien virustarkistus
- X-järjestelmä
- verkko-osoitteiden jako
- tulostinpalvelut.

Palvelin 2 pystytettiin ainoastaan varmuuskopiointia varten. Siihen varmuuskopioidaan kerran viikossa käyttäjien kotihakemistot siltä varalta, että Palvelin 1 hajoaisi tai käyttäjät vahingossa tuhoaisivat tiedostojaan. Palvelin 1 sisältää peilaavan levyohjaimen, joka peilaa kaiken tiedon kahdelle samankokoiselle levylle automaattisesti.

Verkon palvelut toteutettiin niin, että niihin pystytään pääsemään käsiksi työasemista riippumattomasti. Poikkeuksena ovat muutamat palvelut, jotka on tarkoitettu vain muutamasta paikasta käytettäviksi, kuten esimerkiksi palkanlaskenta ja kirjanpito.




Kuvio 2. Toiminnassa oleva uusi verkko.

Uudesta verkkototeutuksesta on kuvan lisäksi olemassa myös interaktiivinen verkkomalli, joka on WWW-selaimella käytettävissä. Mallista ovat nähtävissä verkkolaitteiden ohjeet, niiden hallinta-osoitteet ja fyysinen sijainti. Esimerkki mallista on kuviossa 2.

# 5 Testaus

Testausvaihe oli koko ajan meneillään verkon rakentamisen aikana, kun sovitettiin aluksi vanha ja uusi järjestelmä toimimaan rinnakkain. Aluksi jouduttiin miettimään koneiden nimiä ja palveluita, ettei ilmenisi ristiriitaisuuksia, jotka pahimmassa tapauksessa estäisivät molempien järjestelmien toiminnan.

Muutoksien jälkeen tehtiin aina testejä, että käytettävät palvelut toimivat niin, ettei käyttäjäkunnalla ole valittamista. Aina suurten muutosten jälkeen pidettiin muutaman viikon testaus, jonka suorittivat tavalliset käyttäjät tekemällä omia työtehtäviään. Ongelmatilanteet ratkottiin analysoimalla ongelmat ja korjaamalla ne mahdollisimman välittömästi.

Vanhan verkon purkaminen aloitettiin sen jälkeen, kun kaikki toiminnot oli saatu siirrettyä ja kokeiltua uudessa verkossa. Vanhaa ratkaisua kuitenkin säilytettiin vielä pari

kuukautta niin, että siihen olisi mahdollista siirtyä pienellä vaivalla.

Testauksen loppuvaiheessa käytettiin verkon analysointiohjelmistoja, joilla varmistettiin, etteivät muutokset aiheuta verkkoon häiriöitä. Samalla nähtiin, millaista liikennettä verkossa liikkuu. Tällä testauksella pystyttiin paikallistamaan mahdollisia ongelmakohtia. Eräs ongelmista oli broadcast-liikenne, jota verkossa oli suuria määriä. Ongelma ratkaistiin määrittelemällä asetuksiin WINS-palvelun käyttö kaikille koneille, joilla se vain oli mahdollista.

Verkon pääpalvelinta myös kuormitettiin alkulukuja laskevalla ohjelmalla, jolloin ilmeni ylikuumenemisongelma, joka olisi myöhemmässä vaiheessa havaittuna ollut erittäin ongelmallinen. Aikaisessa vaiheessa huomattuna oli mahdollista parantaa palvelinhuoneen ja palvelimen jäähdytystä häiriöitä aiheuttamatta.

Testausta tehtiin myös normaalissa työympäristössä mittaamalla vasteaikoja, kun käyttäjät tekivät normaaleja työasioitaan. Tämä testaus antaa realistisimman kuvan millaista käyttöä järjestelmän tulee kestää. Tavallisessa käytössä saadaan kuormaa aiheutettua maksimissaan noin kymmenen prosenttia koko järjestelmän prosessorin maksimitehosta. Muistin käyttö maksimissaan nousee noin pariin prosenttiin, vaikka työasemissa olisi useita raskaita ohjelmistoja auki samanaikaisesti. Testauksien tuloksista nähtiin selvästi, että verkossa ja palvelimessa on varsin paljon kapasiteettia ja nykyisellä konemäärällä ei voida saada kaikkea käytettyä. Tuloksista voidaan keskimäärin laskea, että konemäärä voi kaksinkertaistua eikä siltikään pitäisi olla suorituskykyongelmia ainakaan normaalissa työkäytössä.

# 6 Käytettävät ohjelmistot

Palvelimen 1 toteutukseen käytettiin seuraavia ohjelmistoja. Perusjärjestelmäksi asennettiin 64-bittinen Gentoo-Linux, joka tarkoittaa, että koko asennus tehtiin käsin ja kaikki ohjelmistot käännettiin portage-työkalulla, minkä ansiosta niistä saatiin erittäin tehokas järjestelmä. Järjestelmän etuna oli lisäksi se, ettei ole olemassa versio-

numerointia, vaan koneeseen voidaan nyt ja tulevaisuudessa asentaa halutut ohjelmistot. Asennusprosessi oli erittäin pitkä, ja vei useita päiviä ennen kuin järjestelmä saatiin edes käynnistymään. Tarkempi kuvaus asennuksesta löytyy Gentoo-Handbookista, joka on saatavissa useilla eri kielillä [3].

Verkko-osoitteet jaetaan DHCP-protokollalla, joka tarjoaa myös X-päätteille tarvittavat tiedot alkulatausta varten ja päätteen käynnistämiseksi verkkoon. Täydellinen konfiguraatiotiedosto löytyy liitteestä 1.

Verkon tiedostojen jakoon käytetään Samba-nimistä ohjelmistoa, joka sisältää avoimen lähdekoodin toteutuksen Windows NT -levyjaoista ja lisäksi käyttäjien hallinnan keskitetysti, jolloin jokaisella käyttäjällä on oma käyttäjätunnus ja kotihakemisto Verkkopalvelimella. Jos käyttäjälle ei ole määritelty käyttäjätunnusta, ei hän pääse käyttämään verkkoa. Samba on erittäin monimutkainen palvelinohjelmisto ja sen opetteleminen kunnolla vie muutaman vuoden. Konfiguroinneissa käytettiin apuna kyseistä ohjetta, ja lisäksi jouduttiin tekemään paljon laajennuksia, että saataisiin käyttöön kaikki tarvittavat toiminnot. Samassa yhteydessä kirjoitettiin ohje Samba v3:n käyttöön ja syntyi Samba 3.XX PDC-opas, joka on ilmaisjakelussa [10]. Täydellinen konfiguraatiotiedosto löytyy liitteestä 2.

WWW-välimuistina käytettiin pakollista välimuistia, jonka läpi kaikki WWW-pyynnöt ohjataan ja näin säästetään Internet-yhteyden kaistanleveyttä. Ohjelmistototeutuksena on Squid, ja lisänä on myös Addzapper, jolla mainokset suodatetaan WWW-sivuilta. Konfiguraatio on liitteessä 3.

X-päätejärjestelmän toteutus on tehty LTSP-järjestelmän mukaisesti, ja se koostuu seuraavista komponenteista: XORG Xfree -palvelin grafiikkaa varten ja BOOTP-palvelin, jolla jaetaan osoitteet, asetukset ja käynnistykseen tarvittavan ytimen sijainti päätteille. NFS tarjoaa levyjaot ja ydintiedoston, että järjestelmän on edes mahdollista ladata itsensä verkosta. TFTP siirtää verkosta ydintiedoston, jolla järjestelmä käynnistyy. XINETD on tarkoitettu suojaamaan yhteyksiä niin, että voidaan kätevästi rajoittaa käyttäjän pääsyä tiettyihin verkko-osoitteisiin. Ohjeet asennukseen ovat saatavissa LTSP-sivustolta ja lisäksi Gentoon omasta LTSP-dokumentaatiosta [4]. Osittainen konfiguraatiotiedosto on luettavissa liitteestä 4.

Edellä kuvatut ohjelmistot muodostavat palvelimen pääkokonaisuuden, eli ne ovat tärkeimmät palvelut, joita tarvitaan. Lisäksi palvelimessa on Exim-postinvälitysohjelmisto, johon on integroitu ClamAV-virusskanneri [7]. Tämä järjestelmä tarkistaa ulospäin lähtevän sähköpostin läpinäkyvästi. Tällaisella järjestelyllä voidaan estää liitetiedostojen mukana leviävien virusten toiminta tehokkaasti ilman että käyttäjien tarvitsee tehdä mitään muutoksia sähköpostiasetuksiinsa.

Järjestelmässä on myös mahdollista käyttää reaaliaikaista virustentarkastusta WWW-liikenteelle ja levyjaoille, mutta se tarvitsee käytännössä tehokkaan lisäpalvelimen tämän ominaisuuden toteutukseen ja siksi ominaisuutta ei otettu käyttöön.

Palvelin 2 sisältää vain varmuuskopiointiin tarkoitettavan asennuksen, ja tiedostot siirretään siihen kerran viikossa käyttäen salattua SSH-protokollaa. Palvelin avaa SSH-yhteyden käyttäen julkisen avaimen menetelmää, ja näin saadaan automatisoitua todennus ilman interaktiivisuutta. Sen jälkeen tiedosto ohjataan putken läpi varmuuskopioivalle koneelle.

Lyhyt esimerkki menetelmästä on:

*tar -zcf - /etc ssh tunnus@kone "cat > /polku/tallennettavaan/tiedostoon.tar.gz"*

Menetelmän etuna on se, että varmuuskopioiva palvelin voi sijaita missä päin maailmaa tahansa, koska kaikki tieto menee salatun yhteyden läpi. Rajoittava tekijä on vain Internet-yhteyden kaistanleveys, jos varmuuskopioita pitää tallentaa verkkoon suurempia määriä. Varmuuskopioitavat tiedostot lisäksi nimetään niin, että jokaisessa varmuuskopiotiedostossa on päivämäärä, jolloin varmuuskopio on otettu. Järjestelmä mahdollistaa siis helpon palauttamisen käyttäjien tiedostoista, jos jotain tiedostoja pääsee tuhoutumaan esimerkiksi käyttäjän virheen takia.

Vikatilanteista saadaan raportti sähköpostiin, jolloin ongelmatilanteet pystytään helposti paikallistamaan ja korjaamaan. Tällä tavalla vältytään siltä, että palvelut eivät toimi riittävän hyvin ja toimintahäiriöt havaitaan vasta sitten kun niiden korjaaminen muuttuu

huomattavasti monimutkaisemmaksi.

Kaikkia tarvittavia ohjelmistoja ei ollut saatavilla, joten osa tarvittavista toiminnoista kirjoitettiin itse käyttäen esimerkiksi Bash-komentokieltä. Lisäksi hankalimpia toimenpiteitä yksinkertaistettiin kirjoittamalla komentojonoja, joilla myös aloittelija pystyy tekemään perustoimenpiteet, kuten esimerkiksi käyttäjätunnuksien lisäämisen järjestelmään ja levyjakojen luomisen.

# 7 Työpöytäohjelmistot

Työpöydille asennettiin käyttöön KDE-työpöytäympäristö ja siihen tarvittavat ohjelmistot, kuten Firefox-selain ja Thunderbird-sähköpostiohjelmisto. Tärkeä asia oli, että ohjelmistot olivat mahdollisimman pääosin suomenkielisiä. Työpöydän piti myös näyttää mahdollisimman paljon Microsoft Windows XP:n työpöydän kaltaiselta. Tämä rajoitti ohjelmistojen valintoja jonkin verran, se karsi Gnome-ympäristön ja kevyen Fluxbox-työpöydän pois valinnoista. Kaikissa ohjelmistoissa ei lisäksi voida käyttää uusinta ohjelmistoversiota sen takia, että käännöstyö ei ole vielä riittävän pitkällä.

Toimisto-ohjelmistona on käytössä OpenOffice ja Soikko-oikaisuluku ohjelmisto. Nämä riittävät peruskäyttöön aivan hyvin. Vakioasennuksesta löytyvät lisäksi yleisimmät pikaviestintäohjelmistot ja muutama peli ajanvietteeksi.

Ohjelmistovalinnoissa pyrittiin suppeaan mutta laadukkaaseen valikoimaan, jolla myös tietokoneen käyttötaidot heikosti hallitsevat pystyvät tulemaan toimeen tietokoneen kanssa. Hyvänä esimerkkinä tästä ovat X-päätteet, jotka on suunniteltu Internet-käyttöön ja työpöydällä näkyvät vain välttämättömät kuvakkeet, jolloin hankala valikoiden selaaminen ei ole tarpeen, vaan WWW-selain voidaan nopeasti ja helposti käynnistää suoraan työpöydältä. Ohjelmistot myös valittiin sen kaltaisiksi, että suuria eroja ei olisi siihen nähden, olisiko käytössä Windows- vai Linux-käyttöjärjestelmä.

# 8 X-päätteen toiminta

X-pääte on ns. ”tyhmä” järjestelmä, joka ei käynnistysvaiheessa sisällä muuta tietoa kuin käynnistyslataajan, joka osaa hakea tiedot verkosta. Päätejärjestelmässä on PXE-käynnistyslataajan sisältävä verkkokortti. X-päätteitä voidaan myös ostaa valmiina tuotteina, jolloin hinta on hieman halvinta tietokoneen keskusyksikköä alempi. X-pääte voidaan myös rakentaa käytöstä poistetuista tietokoneista, jotka eivät enää sisällä riittävästi prosessoritehoa nykypäivän käyttöjärjestelmien käyttöön. Tarvitaan vain käytöstä poistettu tietokone ja verkkokortti, jossa on PXE-ominaisuudet. PXE-ominaisuuksien puuttuessa voidaan käyttää myös levykettä, jossa on tarvittava osa käynnistyslataajaa.

Järjestelmän etuna on se, että koneesta saadaan tehtyä äärimmäisen hiljainen, koska sen ei tarvitse sisältää yhtään liikkuvaa osaa. Lisäksi pääte toimii yhtä nopeasti kuin palvelin, johon se on verkkoyhteydellä liitetty. Hajonnut laite lisäksi voidaan korvata uudella ilman asetusmuutoksia, ja tämä helpottaa tietohallinnon työtä.

Ongelmana on, että pääte ei sovellu kunnolla toimintoihin, jotka vaativat liikkuvaa videokuvaa tai äänen käsittelyä, mutta nämä ovat pieniä puutteita hyötyihin verrattuna. Yleensä toimistokäytössä ei tarvita kovinkaan paljon ominaisuuksia, joten tällaiseen käyttöön X-pääte sopii erinomaisesti.

Tekniikka on yleisesti paljon käytetty köyhissä maissa, ja se on leviämässä myös koulukäyttöön sillä saavutettavien etujen vuoksi.

Seuraavaksi esimerkki Linux-järjestelmään käynnistyvästä X-päätteestä:

-XPÄÄTE: virrat päälle ja BIOS-toimintojen käyminen läpi

-XPÄÄTE/BOOTP: kaiutetaan MAC-osoite verkkoon asetuksia varten

-PALVELIN/BOOTP: annetaan verkko-osoite ja käynnistystiedostojen sijainti

-XPÄÄTE/TFTP: ladataan käynnistystiedosto muistiin.

-XPÄÄTE: puretaan käynnistystiedosto muistiin ja alustetaan oheislaitteet.

-XPÄÄTE: NFS-levyjaosta käynnistetään INIT-ohjelmisto ja tarvittavat asetustiedostot luodaan samalla

-XPÄÄTE: avataan yhteys X-palvelimeen

-XPÄÄTE: näytetään X-palvelimelta tulevaa kuvaa

Tämän jälkeen järjestelmä on toimintavalmis ja käyttäjät voivat käyttää päätettä kuten tavallista tietokonetta, mutta prosessointi tapahtuu vain läpinäkyvästi.

# 9 Kustannukset

Linux-pohjaisella ratkaisulla voidaan säästää paljon rahaa, tämä tosin edellyttää korkeaa teknistä osaamista verkkoratkaisuissa. Työn arvoa ei laskettu rahassa, mutta ulkopuolisella tekijällä kustannusarvio olisi ollut noin 20 000 euroa, kun lasketaan keskimääräisesti käytetyt työtunnit ja kerrotaan saatu luku keskimääräisellä tuntihinnalla, joka on noin 90 euroa/h. Työstä ei insinöörityön vuoksi veloitettu, joten kustannukset työn osalta olivat alle sata euroa. Pitää myös muistaa, ettei Windowsilla ole mahdollista toteuttaa tällaisia järjestelmiä ja myös osaavat MSCE-järjestelmäasiantuntijat veloittavat suunnittelusta, konsultoinnista ja toteutuksesta ainakin saman verran kuin itse arvioin tuntihinnaksi. Työn kustannukset pysyvät käytännössä vakioina järjestelmästä riippumatta.

Suurin säästö tuli lisensseissä ja laitteistoissa, sillä käytännössä kaikki talon ohjelmistot ja suurin osa koneista olivat uusimisen tarpeessa. Taulukossa 1 näkyvät kilpailevan järjestelmän lisenssikustannukset, jotka avoimen lähdekoodin tuotteilla vastaavasti ovat 0 euroa. Lisäksi ohjelmistoja on käytössä enemmän kuin taulukossa, jossa on tarjolla vaihtoehdot vain toimihenkilöiden käyttöön.

Taulukko 1. Lisenssikustannukset.

| | A | D | E | G | H |
|---|---|---|---|---|---|
| 1 | Tuote | hinta (euroa) | tarvittava määrä | Kokonaissumma (euroa) | |
| 2 | Windows 2000 advanced server | 5041 | 1 | 5041 | |
| 3 | Office XP professional | 395 | 10 | 3950 | |
| 4 | CAL-lisenssit | 50 | 10 | 500 | |
| 5 | Yhteensä | | | 9491 | |

Kilpailevaa käyttöjärjestelmää varten myös koneet pitäisi uusia, ja niiden kustannusarvio on noin 10 000 euroa. Linux-järjestelmällä käytetään vanhoja tietokoneita, joten kustannus koneiden uusimiseen on 0 euroa. Palvelinjärjestelmät ja aktiivilaitteet olisi jouduttu uusimaan joka tapauksessa, ja niiden kustannukset ovat siis suunnilleen samat.

Windows-järjestelmien lisensointipolitiikka on lisäksi niin hankala, ettei siitä tunnu olevan edes täyttä varmuutta, ja erilaista hinnoittelupolitiikkaa noudatetaan myös eri asiakkaille. Hintavertailu on tehty käyttämällä verkkokauppa.com-hinnastoa ja kysymällä Microsoftilta. Hintavertailu ei myöskään ole kilpailukykyinen siksi, että Linux-järjestelmällä pystytään toteuttamaan tyhmä X-päätejärjestelmä, jollaista ei oikein pysty Windowsilla toteuttamaan, ellei käytetä Citrixiä ja siihen liitettyjä minipäätteitä, jotka ovat kalliita. Lisäksi jokaiselta päätelaitteelta pitää erikseen maksaa ns. CAL-lisenssi, mikä tarkoittaa käytännössä noin 90 euron lisäkustannusta päätettä kohti. Lisäksi oli myös ristiriitaista tietoa siitä, joutuuko CAL-lisenssin maksamaan myös jokaisesta päätteestä, jolla halutaan käyttää Windows-palvelimien tiedostojakoja.

Taulukko 2 kertoo Windows- ja Linux-järjestelmien kustannusten eron, joka on erittäin suuri, koska toisessa järjestelmässä olisi joutunut tekemään suurempia laiteinvestointeja. Vertailuun ei ole sisällytetty työn osuutta, sillä sen vertailu on erittäin hankalaa, joten oletetaan työmäärän ja hinnoittelun olevan suunnilleen samaa luokkaa. Vuosittaisia lisenssikustannuksia ei myöskään ole sisällytetty taulukkoon.

Taulukko 2. Hintavertailu Linuxin ja Windowsin välillä.

| 7 | | Windows (euroa) | Linux (euroa) | Ero (euroa) |
|---|---|---|---|---|
| 8 | Tietokoneet | 10000 | 0 | 10000 |
| 9 | Aktiivilaitteet | 2000 | 2000 | 0 |
| 10 | Ohjelmistot | 9491 | 0 | 9491 |
| 11 | Yhteensä | 21491 | 2000 | 19491 |

Ylläpitokustannuksissa säästyy huomattavasti rahaa, koska keskitetyssä palvelinjärjestelmässä ohjelmistojen päivitykset voidaan tehdä yhdellä koneella ja muutos on heti nähtävissä kaikilla päätejärjestelmissä olevilla koneilla.

Virusten ja haittaohjelmistojen puuttuminen Linux-järjestelmästä lisää myös kustannusetua, sillä ei tarvitse huolehtia virusten ja lisenssien aiheuttamista kustannuksista. Työpöytäympäristön ulkoasun muuttuminen toisaalta aiheuttaa koulutustarvetta henkilöstölle, mutta tämä kyllä pystytään hoitamaan organisaation sisällä, joten se ei tuo ylimääräisiä kustannuksia. Vanhankin järjestelmän päivitys Windows XP -ympäristöön olisi aiheuttanut samanlaista koulutustarvetta, joten voidaan olettaa, että koulutuskustannukset olisivat silti suunnilleen vakioita. Isojen muutoksien tekeminen järjestelmiin tuo aina tiettyjä sivukustannuksia, joiden voidaan laskea kuuluvan tietojärjestelmien ylläpitoon.

## 10 Pohdinta

Uusi verkko on nyt ollut täysin käytössä noin kolme kuukautta eikä suurempia ongelmia ole ollut. Näin mittavassa urakassa on aina tietysti muutaman kuukauden testivaihe, jonka aikana toimintaa hiotaan korjaten ilmenneet puutteet ajan salliessa.

Käyttäjille näkyviä ominaisuuksia ovat olleet ne, että vanhoihin koneisiin on saatu uutta

tehoa, kun on siirrytty käyttämään X-päätejärjestelmää ja vakioidut työpöydät ovat näin käytössä jokaisessa työpisteessä tarvittaessa. Uusi järjestelmä on myös näkynyt nopeampana toimintana, sillä verkkoon on tehty fyysisiä muutoksia ja aktiivilaitteet on saatettu ajan tasalle.

Kustannussäästöt ja luotettava toiminta ovat olleet suurin asia verkon uudistuksessa. Tietohallinto on ollut tyytyväinen, kun verkon tiedostot ovat oikeasti tallessa kahdessa paikassa, peilaavalla levyjärjestelmällä ja sen lisäksi varmuuskopiopalvelimella.

Ongelma on edelleen rahatilanne; kaikkia toteutuksia ei ole voitu tehdä aivan täydellisesti, koska on jouduttu ajattelemaan myös budjettia. Keskihintaisella laitteistolla tosin saavutettiin hyviä tuloksia, kun valittiin oikeat laitteistot. Internet-pohjaisia palveluja ei näillä resursseilla ainakaan vielä voida tarjota. Suurin kynnyskysymys ovat symmetriset verkkoyhteydet, joiden hinta on edelleen satoja euroja kuukaudessa. Tulevaisuudessa tilanne voi muuttua ja ulkopuolelta ostettavat sähköpostipalvelut voidaan tuottaa itsenäisesti.

Kilpailevan järjestelmän tuotteilla kustannukset olisivat olleet yli kaksinkertaiset eikä kaikkia ominaisuuksia ei olisi saatu käyttöön järkevällä hinnoittelulla. Lisäksi yleensä palvelinohjelmistojen lisenssit ovat käyttäjäkohtaisia, eli järjestelmä maksaa siis enemmän, mitä enemmän käyttäjiä on verkossa. Lisäksi vuosittaiset ylläpitokustannukset ja ohjelmistojen päivityksistä aiheutuneet kustannukset tulisivat useamman vuoden ajanjaksolla kalliiksi.

Järjestelmän käyttöiäksi on Linux-puolella laskettu noin viisi vuotta, jonka jälkeen joudutaan uusimaan palvelinpuolen laitteita, mutta työasemat voivat pysyä vakioina tai käyttöön voidaan ottaa muualta jo poistettuja tietokoneita, joita yleensä saadaan erittäin halvalla. Käyttämällä X-päätejärjestelmää on erittäin helppo saada koneista pitkäikäisiä, kun poistetaan suurin osa liikkuvista osista, kuten tallennusmediat ja CD-asemat. Samalla täytetään hyvin valtiohallinnon tietoturvasuositukset, koska aivan kaikki tieto saadaan tallennettua verkkolevyille. Käytöstä poistuvat työasemat on lisäksi turvallista hävittää, koska niihin ei saada tallennettua mitään arkaluontoista tietoa, joka vääriin käsiin joutuneena voisi johtaa oikeusjuttuun ja jopa vankilatuomioihin. Käytettävä laitteisto lisäksi on niin vanhanaikaista, että se ei ole edes varkaiden suosiossa, ja

korvaavaa laitteistoa on helppo saada, koska kaikki asetukset tehdään palvelimessa ja työasema tarvitsee vain kohtuullisen verkkokortin ja näytönohjaimen.

Järjestelmä on pystynyt vastaamaan riittävän tehokkaasti ja vakaasti nykyiseen ympäristöön. Linjakapasiteettia oli tarkoitus nostaa parin kuukauden kuluessa, jolloin hitaan Internet-yhteyden tuomat ongelmat poistuvat verkosta. Ainoastaan OpenOffice-toimistopaketin kanssa on ilmennyt pieniä ongelmia, joiden oletetaan korjaantuvan versioon 2.0, joka julkaistaan marraskuussa 2005.

Kokonaiskuvana työ oli todella haastava. Työstä jouduttiin karsimaan osa dokumentaatiosta, varsinkin konfiguraatiotiedostojen osalta, koska se olisi muuten vienyt satoja sivuja. Dokumentointia olisi pitänyt tehdä enemmän insinöörityön aikana, mutta kiireessä osa työstä meni rutiiniasennuksena ja jälkikäteen joutui pohtimaan, miten asia oikeasti tuli toteutettua.

Prosessina työssä olisi ollut huomattavasti kehittämistä, koska osa työstä tehtiin "tarpeen mukaan" eikä kaikkea suunniteltu aivan täydellisesti. Tämä näkyy siinä, että tarpeet muuttuivat suunnitelluista jossain kohtaa melko rajustikin.

Suunnitteluorganisaation puute oli ehkä suurin ongelma. Järjestelmä vain tehtiin, eikä kommentteja saatu oikeastaan ollenkaan järjestelmän huonoista puolista, vaikka palaute olisi ollut paikallaan. Projektiin olisi pitänyt perustaa projektityöryhmä, jonka kanssa asiat olisi käyty selkeästi läpi selostaen, mitä suunniteltu vaihe tarkoittaa ja mitä kustannuksia se tuottaa, sekä lisäksi havainnollistaen, mitä ratkaisu mahdollistaa. Ammattilaisen ja päättävien ihmisten välinen vuorovaikutus olisi ollut tärkeä asia. Tämä puute tosin suurelta osin johtuu siitä, että insinöörityö tuli kuvioihin järjestelmän keskivaiheilla, järjestelmää ei alun perin ollut suunniteltu insinöörityönä vaan Linux-kerhon kanssa yhteistyönä. Insinöörityön aloittaminen puolivälistä hankaloitti sitä, että asiaa ei voitu tehdä täysin johdonmukaisesti alusta lähtien, kuten yleensä insinöörityöt tehdään. Tämä ei silti aiheuttanut suuria ongelmia ratkaisun toteuttamiseen. Ainoastaan tämä vaikeutti hieman prosessia, mutta se toisaalta tai myös realistisemman kuvan tosielämän asioiden hoidosta. Tämä myös antoi itselleni enemmän vastuuta ratkaisun toimivuudesta.

Toinen ongelma oli asiakkaan tietämättömyys ratkaisusta. Kyseistä asiaa olisi voinut helposti verrata vaikka talon rakennukseen. Talo yleensä tehdään loppuun asti eikä sitä voida jättää esimerkiksi ilman kattorakenteita. Tietoverkko taas käsitetään enemmän komponentteina, ja välillä oli ongelmana saada ihmiset käsittämään, että myös tietoverkko pitää rakentaa loppuun eikä sitä voida jättää vaillinaiseksi ilman että sen käyttötarkoitus kärsii. Suurin ongelma oli asian konkretisoiminen. Talosta näkee helposti aloittelijakin, että katto on tarpeellinen osa, mutta tietoverkon tai tietojärjestelmän puuttuvia asioita eivät aloittelijat pysty hetkessä havainnoimaan.

Tietämättömyysongelma on yleinen hallinto- ja rahapuolella: tingitään ratkaisuista tietämättä niiden kokonaisvaikutuksia ja mahdollisia ongelmatilanteita. Tuli selväksi, että pitäisi tehdä jonkinlaista tietopakettia päättäville ihmisille, jotta saataisiin tietojärjestelmät yhtä selkeäksi kuin esimerkiksi rakennettavan talon perusteet. Tietotekninen ala on vielä niin nuori, että yleisetkin asiat saattavat olla hämärän peitossa ja on suositeltavaa tehdä tietoverkoista sellaisia pakettiratkaisuja, jotka ostetaan komponenttien sijasta kokonaisuuksina.

Seuraava huolestuttava asia olivat ylläpitokustannukset. Tietojärjestelmistä päättävien ihmisten tiedossa ei yleensä ole, että järjestelmiin tulee varata säännöllinen ylläpitorahoitus, sillä muuten käy kuin talolle, josta puuttuvat säännölliset huoltotoimenpiteet. Ylläpitämätön ratkaisu rapistuu muuten sille tasolle, että uuden rakentaminen tulee halvemmaksi kuin vanhan korjaaminen. Tähän tekijään vaikuttaa organisaation luonne, ja se vain on otettava huomioon tässä tapauksessa. Toisaalta järjestelmässä ei ole mitään täysin kriittisiä komponentteja eikä ihmishenkiä ole vaarassa, vaikka verkko ei saavuttaisikaan kaupallisten organisaatioiden tasoisia vaatimuksia ylläpidon nopeudessa. Ohjelmistopuolella pitäisi olla saatavissa helposti käytettäviä ohjelmistoja, joista on karsittu turhat asiat pois. Suomennoksissa on huomattavasti tekemistä, että saadaan asiat sille tasolle, millä niiden pitää olla. Tietokoneen pitäisi olla niin helppokäyttöinen ja yksinkertainen, että asiat voidaan tehdä helposti ilman pelkoja koneen rikkoutumisesta tai kaatumisesta. Yksinkertaisetkin vikatilanteet saattavat pelottaa aloittelevaa tietokoneen käyttäjää ja häiritä oppimista tai pahimmassa tapauksessa aiheuttaa pelkoa tietokonetta kohtaan niin paljon, ettei siihen haluta koskea enää seuraavaa kertaa.

Linuxin työpöytäsoveltuvuudesta saatiin myös uusi käsitys ja huomattiin, että se on varsin hyvin sillä tasolla, että tavallinen ihminen sitä pystyy käyttämään päivittäisiin työaskariinsa, jos uuden järjestelmän tuomat ennakkoluulot pystytään voittamaan. Teknisessä mielessä insinöörityö oli onnistunut, vaikka en voinutkaan täysin toteuttaa kaikkia ideoitani. Työ oli silti yksi laajimmista Linux-järjestelmistä Itä-Suomen alueella. Tulevaisuudessa voidaan olettaa, että muutkin organisaatiot siirtyvät päätejärjestelmiin, koska se tuo paljon etuja ja mahdollistaa sellaisiakin asioita, joiden toteuttaminen muute olisi erittäin kallista.

Tulevaisuudessa järjestelmää voidaan laajentaa tarpeiden mukaan, jos rahoitus saadaan järjestettyä. Eräs mietinnän alla olevista ratkaisuista on siirtää puhelinliikenne käyttämään nettiyhteyksiä, sillä silloin voidaan saada suuria säästöjä puhelinkustannuksissa ja joustavuutta puhelinten sijaintiin, kun talossa on jo valmiina riittävän kattava langaton lähiverkkoratkaisu. Tulevaisuudessa voidaan ehkä myös siirtyä enemmän käyttämään oman verkon palveluja esimerkiksi sähköpostiratkaisuissa, jos nopeat Internet-yhteydet halpenevat lähitulevaisuudessa. Intranetin palveluja voidaan ruveta tarjoamaan Kotikartanoyhdistyksen alaisuudessa toimiville jäsenjärjestöille, jos tarvetta tällaiselle joustavuudelle on.

Järjestelmä on kooltaan erittäin suuri, jos verrataan järjestelmässä olevia säätöjä perusasennukseen. Tämä tuo esille myös sen ongelman, että isojen järjestelmien dokumentointi on erittäin työiästä. Riittävän täydellisen dokumentoinnin tuottaminen tästäkin järjestelmästä olisi lisännyt järjestelmän rakentamiseen kuluvaa aikaa yli puolella vuodella. Insinöörityössä tämä vaatimus on mahdoton toteuttaa.

Järjestelmä jättää vielä vapaaksi tilaa toiselle insinöörityölle, jonka aiheena voisi olla vaikkapa verkon täydellinen dokumentointi ja verkkokuvien tarkkuuden laajentaminen ja ylläpitohenkilöstön kouluttaminen sille tasolle, että verkon kanssa tullaan toimeen ilman ulkopuolista ylläpitoa.

## Lähteet


1 Avoimen lähdekoodin (Open Source) määritelmä. 2005. Verkkodokumentti http://www.free-soft.org/mirrors/www.opensource.org/docs/osd-finnish.php . Luettu 1.3.2005.

2 Firefox FAQ. 2005. Verkkodokumentti. http://koti.mbnet.fidf-faq/general.shtml . Luettu 1.4.2005.

3 Gentoo foundation. Gentoo Handbook. 2003. Verkkodokumentti Gentoo.org. http://www.gentoo.org/doc/en/handbook/handbook-amd64.xml?ful1=1 . Luettu 1.6.2005

4 Gentoo foundation. Gentoo LTSP-guide. 2005. Verkkodokumentti Gentoo.org. http://www.gentoo.org/doc/en/ltsp.xml Luettu 1.7.2005.

5 Koskinen, Asmo. Suomenkielinen OpenOffice. Verkkodokumentti OpenOffice.org. http://fi.openoffice.org/ . Luettu 1.6.2005.

6 Mikä on NAT. 2004. Verkkodokumentti. http://www.finx.org/fi-faq.php#mikaonnat. Luettu 1.5.2004

7 Setting up mailscanner with Exim 4.X 2004. Verkkodokumentti. http://www.flatmtm.com/computer/Linux-EximMailScanner.html . Luettu 1.5.2004

8 Symatec corporation. W32.blaster.Worm. 2004. Verkkodokumentti Symatec Oy. http://securityresponse.symatec.com/avcenter/venc/w32.blaster.worm.html Luettu 1.6.2004

9 Volotinen, Eero. Samba PDC opas. 2005. Verkkodokumentti Eero Volotinen http://pronics.fi/~eero/mirrors/samba3-pdc/Samba%203.XX%20PDC%20OPAS.html . Luettu 1.6.2005

10 Volotinen, Eero. Samba PDC 2 opas. 2004. Verkkodokumentti Eero Volotinen http://pronics.fi/~eero/mirrors/jinux/samba/ . Luettu 1.4.2005

**Liite 1 (3)**

**Asetukset dhcp-palvelulle**

```
# /etc/dhcpd/dhcpd.conf
# Sample configuration file for ISCD dhcpd # with boot menu

ddns-update-style          ad-hoc;
default-lease-time         21600;
max-lease-time             21600;
option subnet-mask          255.255.255.0;
option broadcast-address    192.168.1.255;
option routers             192.168.1.2;
option domain-name-servers 192.168.1.2;
option domain-name          "foo.invalid";

option root-path           "192.168.1.2:/opt/ltsp-4.1/i386";
filename                   "/lts/vmlinuz-2.4.26-ltsp-2";

option log-servers         192.168.1.2;
next-server                192.168.1.2
option netbios-name-servers 192.168.1.2;
use-host-decl-names on;


shared-network WORKSTATIONS {
  subnet 192.168.1.0 netmask 255.255.255.0 {
     range dynamic-bootp 192.168.1.33 192.168.1.230; #use-host-decl-names on;

  host ws71{
     hardware ethernet    00:E0:4C:86:24:21;
     fixed-address        192.168.1.71;
      filename            "/pxe/pxelinux.0"
}
host ws72 {
    hardware ethernet     00:00:F8:75:84:BB;
```

```
    fixed-address          192.168.1.72;
    filename        "/lts/vmlinuz-2.4.26-ltsp-2";

  }
  host ws70
    hardware ethernet    90:12:02:51:0B:4C;
    fixed-address          192.168.1.70;
    filename        "/pxe/pxelinux.0";
  }

  host ws85 {
    hardware ethernet    00:00:F8:75:80:8D;
    fixed-address          192.168.1.85;
    filename        "/lts/vmlinuz-2.4.26-ltsp-2";
}


  host ws80 {
    hardware ethernet    00:40:D0:5A:D5:18;
    fixed-address          192.168.1.80;
    filename        "/pxe/pxelinux.0";
    }

  host ws79 {
    hardware ethernet    00:03 :OD:13 :E2:9F;
    fixed-address          192.168.1.79;
    filename        "/pxe/pxelinux.0";
    }


host ws81 {
    hardware ethernet    00:E0:4C:86:24:21;
    fixed-address          192.168.1.81;
    filename        "/ltsp/vmlinuz-2.4.26-ltsp-2";
}
```

```
  #NORMAL ETHERBOOT HOST



  host wsfoo {
     hardware ethernet     00:0B:6A:6A:D1 :50;
     fixed-address              192.168.1.81;
#    filename          "/lts/vmlinuz.ltsp";
     filename          "/lts/vmlinuz-2.4.21-Itsp-1";

  }

     host ws89 {
     hardware ethernet     00:C0:26:83:38:FF;
     fixed-address              192.168.1.89;
# filename "/lts/vmlinuz.ltsp";
     filename          "/lts/vmlinuz-2.4.21-Itsp-1";

  }


}
}
```

**Liite 2 (5)**

**Samba-asetukset PDC mallin toimialueeseen**.

```
# /etc/samba/smb.conf

[global]

workgroup = KKARTANO

bind interfaces only = Yes

pam password change = Yes

passwd chat = *New*Password* %nln *Re-enter*new*password*%nln

*Password*changed*

username map = /etc/samba/smbusers

unix password sync = Yes

#log level = 1

syslog = 0

log file = /var/log/samba/%m

max log size = 50

#smb ports = 139 445

name resolve order = wins bcast hosts

time server = Yes

printcap name = CUPS
```

show add printer wizard = No

add user script = /usr/sbin/useradd -m '%u'

delete user script = /usr/sbin/userdel -r '%u'

add group script = /usr/sbin/groupadd '%g'

delete group script = /usr/sbin/groupdel '%g'

add user to group script = /usr/sbin/usermod -G '%g"%u'

add machine script = /usr/sbin/useradd -s /bin/false -d /tmp '%u'

shutdown script = /var/lib/samba/scripts/shutdown.sh

abort shutdown script = /sbin/shutdown -c

logon script = scripts\logon.bat

logon path = \\%L\profiles\%U

logon drive = X:

logon home = \\%L\%U

domain logons = Yes

preferred master = Yes

wins support = Yes

#utmp = Yes

#map ad inherit = Yes

printing = cups

veto files = /*.em1/*.nws/*.(*)/

veto oplock files = /*.doc/*.xlsi*.mdb/

unix charset = utf8

dos charset = CP850

client code page = 850

*#* For Samba 3.x. This enables ClamAV on access scanning.

#vfs object = vscan-clamav

#vscan-clarnav: config-file = /etc/samba/vscan-clamav.conf

[homes]

comment = Home Directories

*#valid* users = %S

read only = No

browseable= No

[printers]

comment = SMB Print Spool

path = /var/spool/samba

guest ok = Yes

printable = Yes

use client driver = Yes

default devmode = Yes

browseable = No

[netlogon]

comment = Network Logan Service

path = /var/lib/samba/netlogon

guest ok = Yes

locking = No

[profiles]

comment = Profile Share

path =/home/profiles

read only = No

#profile ads = Yes

[WWW]

guest ok = yes

comment = Intranet sivut

writable = yes

path = /var/www

msdfs proxy = no

```
case sensitive = no

[yhteiset]

comment = Talon yhteiset tiedostot

writable = yes

create mode = 0777

public = yes

path = /home/yhteiset

directory mode = 0777

[mitu]

writeable = yes

delete readonly = yes

locking = no

path = /home/mitu

write list = jutikle,@mitu comment = Testataan

create mode = 770

directory mode = 770
```

**Asetukset www-välimuistille**

```
# /etc/squid/squid.conf

http_port 8080

httpd accel host virtual

httpd_accel_port 80

httpd accel_with_proxy on

httpd_accel_uses_host_header on

hierarchy stoplist cgi-bin ?

acl QUERY urlpath_regex cgi-bin \?

no_cache deny QUERY

cache mem 16 MB

cache_dir ufs /var/cache/squid 1000 16 256

redirect_program /etc/adzapper/squid_redirect

redirect children 10

auth_param basic children 5

auth_param basic realm Squid proxy-caching webserver

auth_param basic credentials ttl 2 hours
```

```
auth_param basic casesensitive off

refresh_pattern ^ftp:        1440 20% 10080

refresh_pattern ^gopher:  1440 0%      1440

refresh_pattem                0                              20%                              4320
acl all src 0.0.0.0/0.0.0.0

ad manager proto cache_object

acl localhost src 127.0.0.1/255.255.255.255



aclto_localhost dst 127.0.0.0/8

SSL_ports port 443 563

acl Safe_ports port 80        # http

acl Safe_ports port 21        # ftp

acl Safe_ports port 443 563 # https, snews

acl Safe_ports port 70        # gopher

ad Safe_ports port 210      # wars

acl Safe_ports port 488      # gss-http

ad Safe_ports port 591      # filemaker

ad Safe_ports port 777      # multiling http

ad Safe_ports port 901      # SWAT
```

ad purge method PURGE

ad CONNECT method CONNECT

http access allow manager localhost

http access deny manager

http access allow purge localhost

http access deny purge

http access deny ! Safe_ports

http access deny CONNECT I SSL_ports

acl our_networks src 192.168.1.0/24 192.168.2.0/24

http access allow our networks

http access allow localhost

http access deny all

http reply access allow all

icp access allow all

visible_hostname uguru

coredump_dir /var/cache/squid

**Liite 4 (2)**

**Asetukset terminaalipäätteille**

```
# /opt/Itsp/etc/lts.conf

# Copyright (c) 2003 by James A. McQuillan (McQuillan Systems, LLC)

# This software is licensed under the Gnu General Public License.

# The full text of which can be found at http://www.LTSP.org/license.txt

# Config file for the Linux Terminal Server Project (wwwitsp.org)

[Default]

SERVER       = 192.168.1.2

XSERVER      = auto

X_MOUSE_PROTOCOL = "PS/2"

X_MOUSE_DEVICE = "/dev/psaux"

X_MOUSE RESOLUTION = 400

X_MOUSE BUTTONS = 3

USE_XFS =   N

SCREEN_01  = startx

XkbLayout = "fi"

X_COLOR_DEPTH = 24
```

[ws70]

PRINTER_0_DEVICE=/dev/lp0

PRINTER_0_TYPE = P

[ws85]

PRINTER_0_DEVICE=/dev/lp0

PRINTER_0_TYPE = P

[ws72]

PRINTER_0_DEVICE=/dev/lp0

PRINTER_0_TYPE = P

[ws73]

PRINTER_0_DEVICE=/dev/lp0

PRINTER_0_TYPE = P

[ws90]

PRINTER_0_DEVICE=/dev/lp0

PRINTER_0_TYPE=P

**Liite 5 (10)**

**Asetukse sähköpostin virustarkastukseen**

#/etc/exim/exim.cnf

*##################################################*
*#################* ROUTERS CONFIGURATION

Specifies how addresses are handled

*##################################################*
*#######0########* # THE ORDER IN WHICH THE ROUTERS ARE DEFINED IS IMPORTANT!

# An address is passed to each router in turn until it is accepted. #

*##################################################*
*##################*

begin routers

# This router routes to remote hosts over SMTP by explicit IP address,

# when an email address is given in "domain literal" form, for example,

# <user@[192.168.35.64]>. The RFCs require this facility. However, it is

# little-known these days, and has been exploited by evil people seeking

# to abuse SMTP relays. Consequently it is commented out in the default

# configuration. If you uncomment this router, you also need to uncomment

# allow_domain_literals above, so that Exim can recognize the syntax of

# domain literal addresses.

# domain literal:

# driver = ipliteral

# domains = ! +local domains # transport = remote_smtp

# This router routes addresses that are not in local domains by doing a DNS

# lookup on the domain name. Any domain that resolves to 0.0.0.0 or to a

# loopback interface address (127.0.0.0/8) is treated as if it had no DNS

# entry. Note that 0.0.0.0 is the same as 0.0.0.0/32, which is commonly treated

# as the local host inside the network stack. It is not 0.0.0.0/0, the default

# route. If the DNS lookup fails, no further routers are tried because of # the no more setting, and consequently the address is unrouteable.

#dnslookup:

# driver = dnslookup

# domains = ! +local domains

# transport = remote_smtp

# ignore_target_hosts = 0.0.0.0: 127.0.0.0/8

# no_more

smart route:

driver=manualroute

transport=remote_smtp

route_list=* smtp.kolumbus.fi byname

# The remaining routers handle addresses in the local domain(s).

.# This router handles aliasing using a linearly searched alias file with the

# name /etc/mail/aliases. When this configuration is installed automatically,

# the name gets inserted into this file from whatever is set in Exim's

# build-time configuration. The default path is the traditional /etc/aliases.

# If you install this configuration by hand, you need to specify the correct

# path in the "data" setting below.

##### NB You must ensure that the alias file exists. It used to be the case ##### NB that every Unix had that file, because it was the Sendmail default. ##### NB These days, there are systems that don't have it. Your aliases ##### NB file should at least contain an alias for "postmaster".

# If any of your aliases expand to pipes or files, you will need to set

# up a user and a group for these deliveries to run under. You can do

# this by uncommenting the "user" option below (changing the user name

# as appropriate) and adding a "group" option if necessary. Alternatively, you

# can specify "user" on the transports that are used. Note that the transports

# listed below are the same as are used for .forward files; you might want # to set up different ones for pipe and file deliveries from aliases.

system_aliases:

driver = redirect

allow fail

allow defer

data = $lookup{$local_part}lsearch{{/etc/mail/aliases}}

# user = exim

file transport = address_file

pipe_transport = address_pipe

# This router handles forwarding using traditional .forward files in users' # home directories. If you want it also to allow mail filtering when a forward # file starts with the string "# Exim filter" or "# Sieve filter", uncomment # the "allow filter" option.

# If you want this router to treat local parts with suffixes introduced by "-"

# or "+" characters as if the suffixes did not exist, uncomment the two local_

# part_suffix options. Then, for example, xxxx-foo@your.domain will be treated

# in the same way as xxxx@your.domain by this router. You probably want to make

# the same change to the localuser router.

# The no verify setting means that this router is skipped when Exim is

# verifying addresses. Similarly, no_expn means that this router is skipped if

# Exim is processing an EXPN command.

# The check ancestor option means that if the forward file generates an # address that is an ancestor of the current one, the current one gets # passed on instead. This covers the case where A is aliased to B and B # has a .forward file pointing to A.

# The three transports specified at the end are those that are used when # forwarding generates a direct delivery to a file, or to a pipe, or sets # up an auto-reply, respectively.

userforward:

driver = redirect

check local user

# local_part_suffix = +* : -*

# local_part suffix optional

file = $home/.forward

# allow filter

no_verify

no_expn

check ancestor

file_transport = address file

pipe transport = address_pipe

reply_transport = address reply

# This router matches local user mailboxes. If the router fails, the error # message is "Unknown user".

# If you want this router to treat local parts with suffixes introduced by "-" # or "+" characters as if the suffixes did not exist, uncomment the two local # part_suffix options. Then, for example, xxxx-foogyour.domain will be treated # in the same way as xxxx@your.domain by this router.

localuser:

driver = accept

check_local_user

# local_part_suffix = +* : -*

# local_part suffix optional

transport = local delivery

cannot_route message = Unknown user

*########################################################*
*###############* TRANSPORTS CONFIGURATION

*########################################################*
*######################*

ORDER DOES NOT MATTER

# Only one appropriate transport is called for each delivery. #

*########################################################*
*#############*

# A transport is used only when referenced from a router that successfully # handles an address.

begin transports

# This transport is used for delivering messages over SMTP connections.

remote_smtp: driver = smtp

# This transport is used for local delivery to user mailboxes in traditional # BSD mailbox format. By default it will be run under the uid and gid of the # local user, and requires the sticky bit to be set on the /var/mail directory. # Some systems use the alternative approach of running mail deliveries under a # particular group instead of using the sticky bit. The commented options below # show how this can be done.

local_delivery:

driver = appendfile

# file — /var/mail/$local_part

directory = /home/local_part/.maildir

maildir_format

delivere_date_add

envelope_to_add

return_path_add

# group = mail

# mode = 0660

# This transport is used for handling pipe deliveries generated by alias or # .forward files. If the pipe generates any standard output, it is returned # to the sender of the message as a delivery error. Set return_fail_output

# instead of return_output if you want this to happen only when the pipe fails # to complete normally. You can set different transports for aliases and # forwards if you want to - see the references to address_pipe in the routers # section above.

address_pipe: driver = pipe return_output

# This transport is used for handling deliveries directly to files that are # generated by aliasing or forwarding.

address file:

driver = appendfile delivery_date_add envelope_to_add return_path_add

# This transport is used for handling autoreplies generated by the filtering # option of the userforward router.

address_reply: driver = autoreply

*##########################################################################W#*

#  RETRY CONFIGURATION

*##########################################################################* begin retry

# This single retry rule applies to all domains and all errors. It specifies # retries every 15 minutes for 2 hours, then increasing retry intervals,

# starting at 1 hour and increasing each time by a factor of 1.5, up to 16

# hours, then retries every 6 hours until 4 days have passed since the first # failed delivery.

# Address or Domain Error Retries

***** F,2h,15m; G,16h,lh,1.5; F,4d,6h

*################################################## ################* REWRITE CONFIGURATION

*################################################## ################*

# There are no rewriting specifications in this default configuration file. begin rewrite

*################################################## ################* AUTHENTICATION CONFIGURATION

*##########################################iffl########### ################*

# There are no authenticator specifications in this default configuration file.

begin authenticators

*################################################## ################* CONFIGURATION FOR local scan()

*################################################## ########W########*

# If you have built Exim to include a local_scan0 function that contains

# tables for private options, you can define those options here. Remember to

# uncomment the "begin" line. It is commented by default because it provokes

# an error with Exim binaries that are not built with LOCAL SCAN HAS OPTIONS

_

# set in the Local/Makefile.

# begin local_scan