LAUREA AMMATTIKORKEAKOULU Uuden edellä

Virtuaalisen palvelinympäristön toteuttaminen tietoliikennelaboratoriolle

Kettunen Pauli, Kuntu Anton

2012 Leppävaara

Laurea-ammattikorkeakoulu
Laurea Leppävaara

Virtuaalisen palvelinympäristön toteuttaminen tietoliikennelaboratoriolle

Kettunen, Pauli & Kuntu, Anton
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Toukokuu, 2012

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

Kettunen, Pauli & Kuntu, Anton

Virtuaalisen palvelinympäristön toteuttaminen tietoliikennelaboratoriolle

Vuosi 2012 Sivumäärä 53

Tämän opinnäytetyön tarkoituksena on ollut toteuttaa Leppävaaran Laurean tietoliikenne laboratoriolle virtuaalinen palvelinympäristö. Palvelinympäristön toteutuksessa on kustannussyistä päädytty käyttämään xen-virtualisointitekniikkaa.

Virtuaalisessa palvelinympäristössä on testattu erilaisia palvelinohjelmistoja, joista on valittu käytännöllisin ja parhaaksi todettu vaihtoehto. Opinnäytetyössä pohditaan myös mahdollisuutta soveltaa pilvipalveluita tietoliikennelaboratoriossa.

Opinnäytetyöprojekti on toiminnallinen. Siksi prosessin kulku sekä käytännön osuudet on myös dokumentoitu.

Projektin palvelinympäristöön liittyvä dokumentaatio käsittää xen-virtualsointitekniikan sekä siihen liittyvät hallinnointityökalut. Lisäksi virtuaaliympäristössä toimivien front end - ja back end -palvelimien toiminta sekä niiden konfiguraatioasetukset on dokumentoitu. Tietoturvaan liittyviä seikkoja pohditaan niin ikään ylläpitäjien kuin loppukäyttäjienkin näkökulmista.

Avainsanat Virtuaalinen palvelinympäristö, Xen-virtualisointi, Pilvipalvelut, Front end - ja Back end -palvelin

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

Abstract

Kettunen, Pauli & Kuntu, Anton

Implementation of a virtual server environment for a telecommunications laboratory

Year 2012 Pages 53

The purpose of this thesis has been to implement a virtual server environment for the telecommunications laboratory of Laurea Leppävaara. The cost reasons have led to the use of xen virtualization technology as the server environment.

The virtual server environment has been tested with a variety of server software, after which the most efficient and effective "best practise" choice has been made. The possibility of using cloud services in the telecommunications laboratory was also discussed in the thesis.

As this thesis is practise-based, the progress of the project and practical sections are documented in it.

The documentation of the project consists of xen virtualization technology and management tools related to the virtual server environment. In addition, the usage of front- and back-end servers operating in the virtual environment and their configuration settings are documented. In this thesis the data security is considered from administrators' and end users' point of view.

Keywords A virtual server environment, Xen virtualization technique, Cloud services, Front end and Back end servers

Sisällys

1 Johdanto

Palvelimien virtualisoimisella tarkoitetaan sitä, että yksi palvelinkone jaetaan useisiin erillisiin virtuaalikoneisiin. Jokaiseen virtualikoneeseen voidaan asentaa oma käyttöjärjestelmä ja tarvittavat sovellukset. Laitteiston ja virtuaalipalvelinten välille rakennetaan ohjelmistollisesti niin sanottu ”virtuaali kerros”, jonka avulla voidaan varata resursseja tarpeen mukaan eri sovelluksille. Tämän prosessin ansiosta fyysisten laitteiden määrä vähenee huomattavasti ja käytössä olevien laitteiden käyttöaste saadaan hyödynnettyä. (Heino 2010, 59)

Tämän projektin tarkoituksena on ollut rakentaa virtuaalinen palvelinympäristö Leppävaaran Laurean tietoliikennelaboratoriolle. Jatkossa käytämme tietoliikennelaboratoriosta lyhennettä tl-laboratorio. Tässä raportissa pyritään selventämään mitä kaikkea projektin virtuaalinen palvelinympäristö kokonaisuudessa käsittää.

Projektissa palvelinympäristön palvelinkoneeseen asennettiin Linuxin Centos-käyttöjärjestelmä. Tämän lisäksi palvelinkoneeseen on rakennettu virtuaalinen kerros, jota hallinnoidaan avoimeen lähdekoodiin perustuvalla xen-hypervisorvirtualisointiohjelmalla. Virtuaalisessa kerroksessa sijaitsee virtuaaliset palvelinkoneet, joiden asentamisvaiheet on dokumentoitu kohta kohdalta. Asennusohjeet on kuvattu tarkemmin liitteen yksi mukaisesti. Projektin toimeksiantajalle on laadittu erillinen dokumentaatio, joka sisältää palveliympäristön käyttäjätunnukset, ip-osoitteet ja teknisen tiedon osuuden. Kyseistä dokumentaatiota ei julkaista tässä raportissa tietoturvaseikkojen takia.

Projektissa kerrotaan yleisesti virtuaalipalvelimiin liittyvistä pilvipalveluista sekä niiden luokittelusta, jonka jälkeen dokumentointi keskittyy tarkemmin virtualisointiin, virtuaalisen palvelinympäristön toteutukseen, xen-verkkoliitäntöihin, xen-hypervisorin hallinnointiohjelmistoihin. Projektissa pohditaan myös muutamalla esimerkillä ylläpitotehtäviä.

2 Tavoitteet

Tämän projektin tavoitteena on ollut toteuttaa Laurean tl-laboratoriolle virtuaalinen palvelinympäristö. Toimeksiantaja on halunnut, että Laurean tl-laboratoriossa olisi virtuaalinen palvelinympäristö, jossa henkilökunta, harjoittelijat ja opiskelijat voisivat tutustua palvelinympäristön toimintaan. Projektissa on laadittu dokumentaatio, joka käsittelee virtuaalisen palvelinympäristön eri osa-alueita.

Virtuaalinen palvelinympäristö tarvitsee ylläpitäjiä toimiakseen, minkä takia projektin eri vaiheet, virtuaalikoneen hallintaohjelmistot ja konfiguraatiotiedostot on myös dokumentoitu. Projektin tavoitteena on ollut myös pohtia eri pilvipalvelumallien soveltamista palvelinympäristön jatkokehityksen kannalta.

3 Kohdeorganisaatio

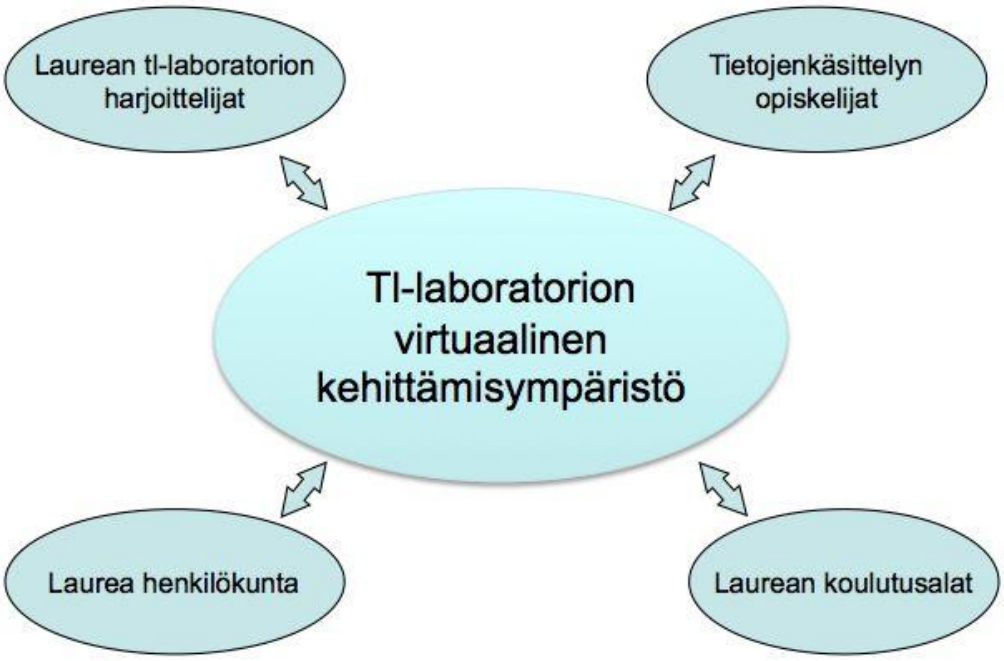
Laurea on Ammattikorkeakoulu, joka toimii organisaationa laajalla Uudenmaan alueella. Laureassa on noin 8000 opiskelijaa, joista yli 1000 on aikuisopiskelijoita. Henkilöstöä on noin 600. Laurealla on yhteensä seitsemän paikallisyksikköä, joista Leppävaaran ja Keravan paikallisyksiköissä voi suorittaa tietojenkäsittelyn koulutusohjelman. Laurean tehtävänä on toimialueidensa paikallisyksiköiden kehittäminen.

Laureassa koulutusohjelmia on mahdollista suorittaa sekä Suomen että Englannin kielellä. Laurean arvoihin kuuluu; opiskelija- ja asiakaskeskeisyys, luotettavuus, yhteisöllisyys, avoimuus, ja yhdessä tekeminen, sosiaalinen vastuullisuus sekä innovatiivisuus. (Laurea fakta 2011)

Tämän projektin toimeksiantajana toimii Leppävaaran toimipisteen tietojenkäsittelyn koulutusohjelman opettaja Riku Salmenkylä. Projekti on toteutettu tl-laboratorion tiloissa.

4 Projektin kohderyhmät

Tärkeimmät kohderyhmät ovat Laurean tl-laboratorion henkilökunta, harjoittelijat, tietojenkäsittelyn opiskelijat sekä mahdollisesti muut Laurean koulutusalat. Edellämainitut kohderyhmät ovat vaikuttaneet tässä projektissa omien koulutusjaksojen sekä hankkeiden kautta. Toiminnan jatkuvuuden kannalta on tärkeätä, että virtuaalista palvelinympäristöä kehitetään. Projekti on suunnattu pääsääntöisesti tietojenkäsittelyn opiskelijoille projektin luonteen vuoksi. Virtuaalista palvelinympäristöä voivat käyttää myös muiden koulutusalojen opiskelijat omissa hankkeissaan tl-laboratorion määrittämien ylläpitäjien avustuksella. Projektissa on otettu huomioon erityisesti tietojenkäsittelyn opiskelijoiden tarpeet, koska tarkoitus on, että he pääsevät käyttämään virtuaalista palvelinympäristöä. Seuraavassa kuviossa kaksi havainnollistetaan tl-laboratorion virtuaalista palvelinympäristöä kehitysympäristönä sekä kohderyhmiä, jotka ovat vaikuttaneet projektissa.



Kuvio 1: Tl-laboratorion virtuaalinen kehittämisympäristö

5 Virtuaalinen palvelinympäristö

5.1 Virtualisointi

Innovaationa virtualisointi ei ole uusi. Erilaisia virtualisointitekniikoita on kehitetty jo 1960-luvulla IBM:n CP-40- ja M44/44X-laitteistojen yhteydessä. CP-40-laitteen CP/CMS-käyttöjärjestelmässä jokaiselle käyttäjälle voitiin luoda oma erillinen virtuaalikone. IBM toi ensimmäisen kaupallisen virtualisoidun koneen markkinoille vuonna 1972. Tämän jälkeen, melkein nykypäivään asti, virtualisointia ei ole osattu hyödyntää kaupallisesti. Vasta viimeisten kymmenen vuoden aikana virtualisointi on muun muassa teknologian, VMwaren ja muutaman muun toimijan ansiosta, tullut vakiopiirteeksi moneen käyttöjärjestelmään ja useimpien laitteistojen päälle. Alla on listattuna yleisiä virtualisointi ratkaisuja:

-vSphere (VMware)
-Hyper-V (Microsoft)
-Xen (Ilmainen, GPL-lisenssin alla)
-z/VM (IBM AIX:n virtualisointi)
-Integrity Virtual Machines (HP)
-Containers (joskus myös “zonet”, Solariksen virtualisointi) (Heino 2010, 59).

Virtualisoinnin hyötyihin voidaan laskea, virtuaalikoneiden helppo hallinta, sillä ei välttämättä tarvitse olla kuin yksi fyysinen laite. Vähäisen laitemäärän vuoksi, virtualisointi on myös kustannustehokasta sekä luontoystävällistä. Samalla palvelinympäristössä on vähemmän vioittumisherkkiä laiteita, jotka vaatisivat huoltotoimenpiteitä. Muihin hyötyihin voidaan laskea varmuuskopiointi, vikatilanteista elpyminen sekä virtuaalikoneen tuhoaminen hallintakonsolissa. (Heino 2010, 59)

5.1.1 Xen-hypervisor

Xen-virtualisointi on avoimeen lähdekoodiin perustuva virtualisointitekniikka. Xen-hypervisor on saatavilla erilaisille alustoille kuten x86, x86_64, IA64, ARM ja muille mikroprosessori arkkitehtuureille. Xen-hypervisor tukee useita käyttöjärjestelmiä kuten Windows, Linux, Solaris ja BSD-käyttöjärjestelmä versioita.

Xen-hypervisor on yksi teollisuuden johtavista virtualisointitekniikoista, sitä käytetään monissa pilvipalveluissa (cloud services) ja webhotelli (web hosting) palveluissa. Tunnettuja yrityksiä, jotka käyttävät xen-hypervisor ohjelmistoa ovat mm. Amazon web services ja Racspace hosting.

Xen-hypervisorista puhuttaessa käytetään yleensä nimitystä xen-virtualisointialusta. Xen-virtualisointialusta tai niin sanottu hypervisor toimii rajapintana fyysisen laitteen ja käyttöjärjestelmän välissä. Tällä rajapinnalla on pääsy tietokoneen fyysisiin laitteisiin ja resursseihin ja sen tarkoituksena on jakaa näitä resursseja vieraskoneille. Xen-virtualisoinnissa vieraskoneilla tarkoitetaan hypervisor rajapinnan päällä ajettavista virtuaalikoneista. (Xen virtualisointi 2011)

5.1.2 Domain 0

Xen virtualisoinnissa on kahdentyyppisiä vieraskoneita ”Domain 0” ja ”Domain U”, näistä käytetään myös lyhennettä dom0 ja domU. Xen-virtualisoinnissa Domain 0 on hallintakone, jolla päästään suoraan käsiksi fyysisen laitteen resursseihin ja sillä myös hallinnoidaan muita vieraskoneita. ”Domain U” vieraskoneiden hallinta siis tapahtuu Domain 0:ssa, joka on yhteydessä xen rajapintaan. (How does xen work 2011)

5.1.3 Domain U

Toisin kuin ”Domain 0” koneella, ”Domain U” koneilla ei ole suoraa pääsyä fyysisen laitteen resursseihin, vaan kaikki resurssipyynnöt menevät aina xen-hypervisor rajapinnan kautta. ”Domain U” koneita on kahden tyyppisiä, paravirtualisoituja ja täysinvirtualisoituja.

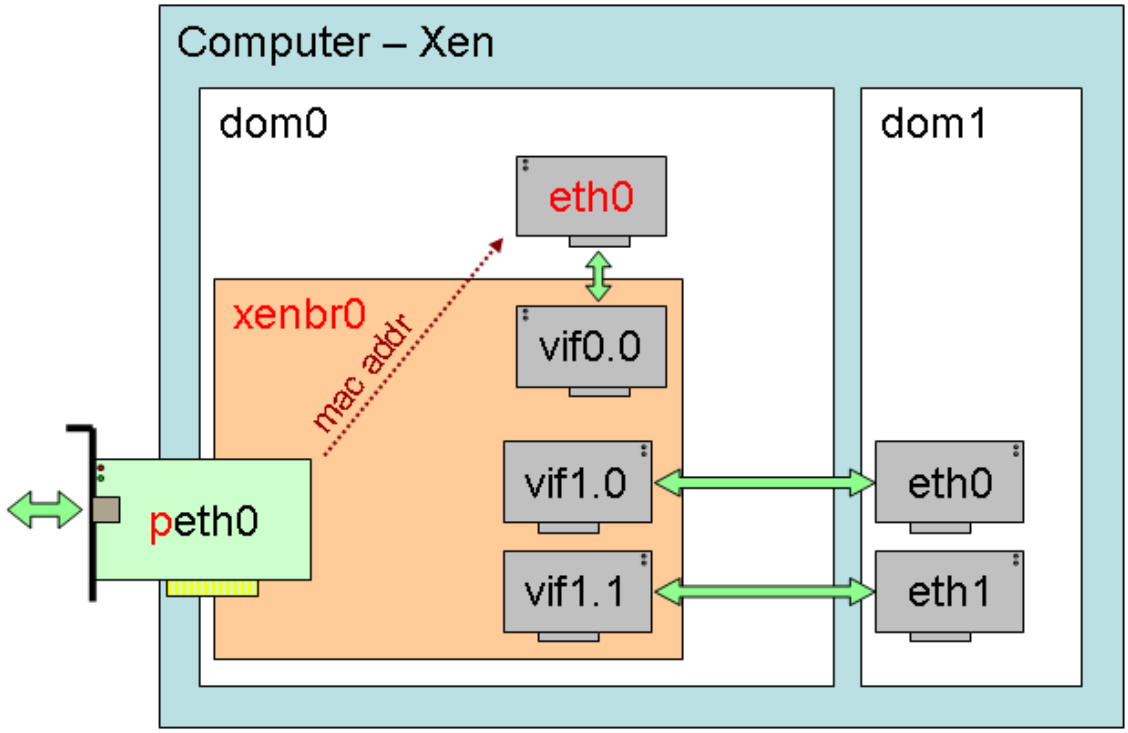
Paravirtualisoitu ”Domain U” tukee muokattuja käyttöjärjestelmiä kuten Linux, FreeBSD, Solaris ja muita Unix-pohjaisia käyttöjärjestelmiä. Täysvirtualisoidulla ”Domain U:lla” voi ajaa myös Windows käyttöjärjestelmiä. (How does xen work 2011)

5.1.4 Xen verkkoliitännät

Xen hypervisor arkitehtuurissa verkkokortteja käytetään jaettuna resurssina. Tämä tarkoittaa sitä, että fyysisen koneen verkkokortteja voidaan käyttää useammassa virtualisoidussa isäntäkoneessa jaettuna resurssina. Xen arkitehtuurissa verkkoliitäntä skenarioita on kahdenlaisia, siltaava yhteys sekä reititetty yhteys. Tämän projektin aikana olemme käyttäneet siltaavaa yhteyttä sen yksinkertaisuuden vuoksi.

5.1.5 Siltaava yhteys

Virtuaalipalvelimia luotaessa ne voidaan liittää verkkoon käyttämällä siltausmenetelmää, jossa virtuaalikoneet käyttävät fyysisen koneen verkkokorttia jaettuna resurssina. Siltaavassa yhteydessä luodaan xenbr0–silta, joka toimii fyysisen kytkinlaitteen tavoin. Fyysinen verkkokortti ajetaan alas ja sen mac-osoite kopioidaan virtuaaliseen verkkoliitäntään ”veth0”. Tämän jälkeen fyysisen verkkokortin ”eth0” nimi muutetaan ”peth0:ksi” ja virtuaalinen ”veth0” verkkoliitännän nimi vaihdetaan eth0:ksi. ”Peth0” ja ”vif0.0” liitetään siltaan ”xenbr0” ja tämän jälkeen silta, ”peth0”, ”eth0” ja ”vif0.0” käynnistyvät. (Xen-verkkoliitännät 2011.) Seuraavassa kuviossa yksi havainnollistetaan siltaavaa yhteyttä projektissa sekä alapuolella on esimerkki siltaavan yhteyden scripti-asetuksista.



Kuvio 2: Kuvio siltaavasta yhteydestä

Siltaavan yhteyden scripti-asetukset:

```
script=/etc/xen/scripts/network-bridge
case ${OP} in
        start)
                $script start vifnum=0 bridge=xenbr0 netdev=eth0
                $script start vifnum=1 bridge=xenbr1 netdev=eth1
                ;;
        stop)
                $script stop vifnum=0 bridge=xenbr0 netdev=eth0
                $script stop vifnum=1 bridge=xenbr1 netdev=eth1
                ;;
        status)
                $script status vifnum=0 bridge=xenbr0 netdev=eth0
                $script status vifnum=1 bridge=xenbr1 netdev=eth1
                ;;
        *)
                echo "Unknown command:${OP}"
                echo 'Valid commands are: start, stop, status'
                exit 1
               ;;
```

5.1.6 Reititetty yhteys

Xen-virtualisointialustalla isäntäkoneen ja virtualikoneen verkkoliikenne voidaan tehdä siltauksen sijaan myös reitityksellä. Hallintakoneen Domain0 käynnistettäessä otetaan samalla käyttöön IP-osoitteen edelleenlähetys. Tämän avulla luodaan reititys dom0:an ja domU:n välille sekä lisätään virtualikoneiden reitit dom0:n reititystauluun. DomU-kone käynnistettäessä, kopioidaan sen IP-osoite isäntäkoneen ”vifX.0” virtuaaliverkkokortille, jonka jälkeen ”vifX.0” verkkoliitäntä käynnistetään ja samassa yhteydessä staattinen reitti on lisätty dom0-hallintakoneen reititystauluun. (Xen-verkkoliitännät. 2011.)

5.2 Front end - ja back end -ympäristöt

Front end -palvelimella tarkoitetaan palvelinlaitetta, joka on verkkosegmentin etupuolella ja vastaanottaa Internetistä tulevia pyyntöjä ja lähettää ne edelleen back end -palvelinympäristöön. Front end -palvelin sijaitsee yleensä verkon ”dmz” eli demilitarisoidulla vyöhykkeellä.

Palvelinympäristössä näillä termeillä tarkoitetaan verkossa sijaitsevia palvelinkoneita, joista front end -palvelin toimii välityspalvelimena. Se vastaanottaa pyyntöjä ja lähettää ne edelleen back end -ympäristöön käsiteltäväksi. Back end -palvelinympäristö käsittää lamp-palvelimet, jotka sijaitsevat verkon sisäpuolella palomuurin takana. Lamp-palvelimet vastaanottaa ja käsittelee front end -palvelimelta tulevat pyynnöt. Alhaalla olevissa esimerkeissä näkyy front end - ja back end -palvelinten konfiguraatioasetukset:

Front end -palvelimen asetukset määritettynä polkuun: /etc/httpd/conf.d/

```
<VirtualHost *:80>
   ServerName oppilas1.sidlabs.fi
   ServerAlias oppilas1.sidlabs.fi
    ProxyRequests Off
    ProxyPreserveHost On
    ProxyPass         / http://192.168.16.233:8081/

<Location />
ProxyPassReverse  /
RequestHeader    unset  Accept-Encoding
</Location>
</VirtualHost>
```

Back end eli LAMP-palvelimen asetukset määritettynä polkuun: /etc/httpd/conf.d/sites.conf/

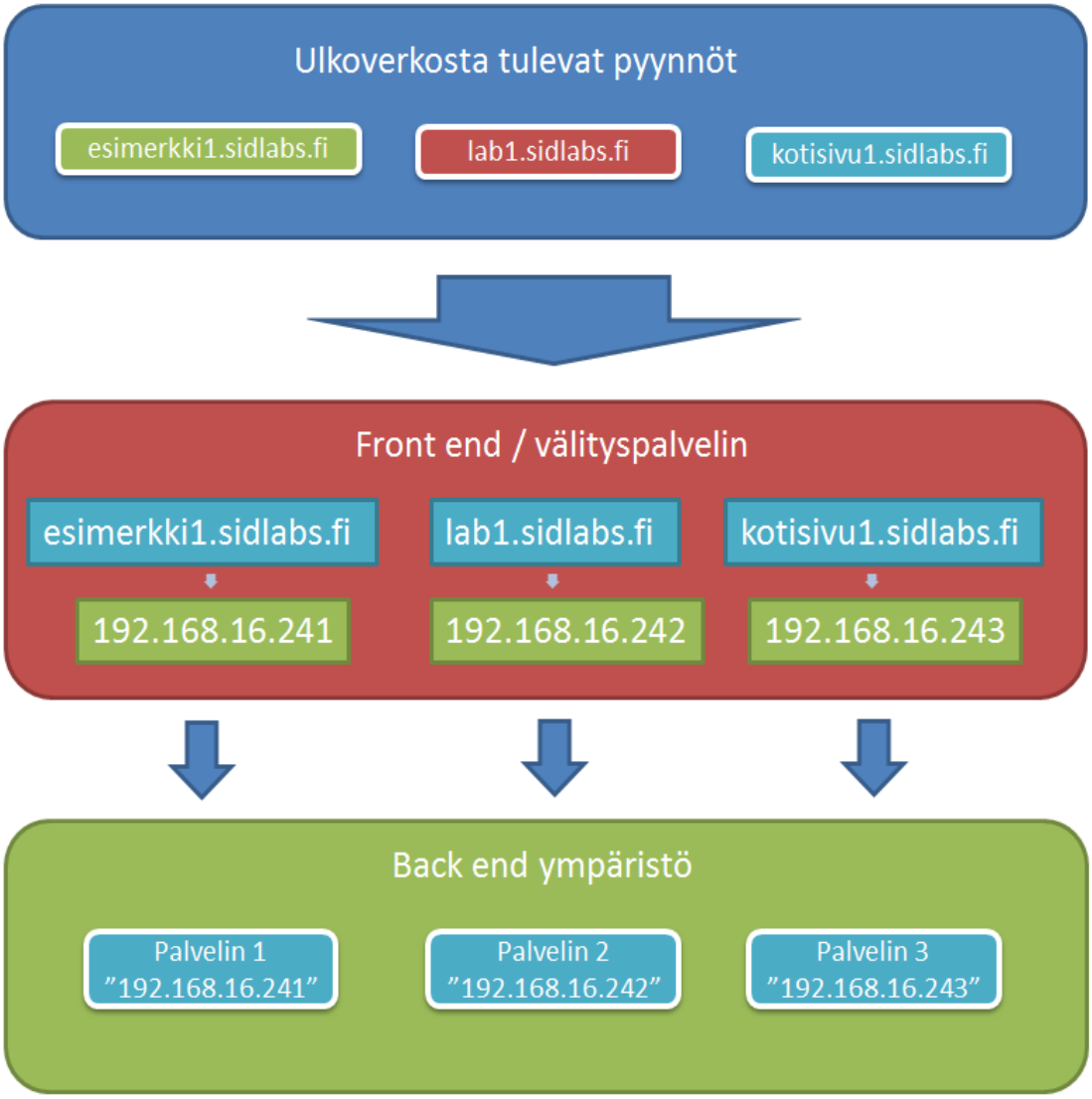
```
Listen 8081
NameVirtualHost 192.168.16.233:8081

<VirtualHost 192.168.16.233:8081>
DocumentRoot "/home/oppilas1/public_html/drupal/"
<Directory "/home/oppilas1/public_html/drupal/">
</Directory>
</VirtualHost>
```

(Ip-ja porttikohtainen näennäispalvelin asetukset. 2011.)

5.3 Wildcard alidomainin toiminta

Virtuaalisen palvelinympäristön välityspalvelimen toiminnallisuuden saavuttamiseksi oli hankittava dns eli nimipalvelu. Välityspalvelimen kannalta nimipalvelun tärkein ominaisuus oli wildcard alidomain tuki. Wildcard mahdollistaa minkä tahansa alidomain osoitteen määrittämisen esimerkiksi mikatahansa.esimerkki.fi. Kohta ”mikatahansa” on alidomainin wildcard. Tämä on tärkeä ominaisuus, mikäli halutaan yhden alidomain nimen alle useita palvelimia tai alisivustoja. Wildcard-tietue liittyy välityspalvelimen toimintaan ja sen kykyyn välittää ulkoverkosta tulevat pyynnöt oikeille back end palvelimille tai back end virtuaalikoneiden näennäispalvelimiin. Näennäispalvelimella tarkoitetaan back end ympäristössä sijaitsevia palvelimia, joissa sivustot sijaitsevat. Seuraavassa kuviossa kaksi havainnollistetaan prosessin kulkua siitä, kuinka välityspalvelin yhdistää wildcard alidomain osoitteen sille määritettyy ip-osoitteeseen.



Kuvio 3: Välityspalvelin ja wildcard

Wildcard osoitteita voi myös käyttää siten, että yhdellä back end -palvelimella on useita näennäispalvelimia ja jokaisella niistä on oma porttinumero sekä siihen liitetty wildcard alidomain osoite. Kyseistä teknistä ratkaisua kutsutaan porttikohtaiseksi näennäispalvelimeksi. Porttipohjaisessa näennäispalvelin ratkaisussa välityspalvelin yhdistää back end -palvelimen porttiin, esimerkiksi wildcardin alidomain osoite oppilas1.sidlabs.fi yhdistetään back end -palvelimen osoitteeseen: 192.168.16.241:8081, jossa opiskelijalla on näennäispalvelin ja sivusto. Samalla back end -palvelimella voi olla siis muiden opiskelijoiden näennäispalvelimia, joilla on eri porttinumero esimerkiksi portti 8082, 8083 ja niin edelleen. Seuraavassa on esimerkki front end -ympäristön konfiguraatioasetuksista, jossa on määritetty niin wildcard alidomain(oppilas1.sidlabs.fi) kuin Proxypass eli back end -palvelimen ip-osoite sekä portin numero, mikä määrittää sivustonpolun.

```
<VirtualHost *:80>
   ServerName oppilas1.sidlabs.fi
   ServerAlias oppilas1.sidlabs.fi
    ProxyRequests Off
    ProxyPreserveHost On
    ProxyPass         / http://192.168.16.241:8081/
```

6 Palvelinympäristön toteutus

Palvelinkoneelle oleellista on, että se sijaitsee turvallisessa paikassa. Turvallisella paikalla tarkoitetaan, että siihen ei pääse ulkopuoliset henkilöt käsiksi. Koska palvelimen tulee toimia ongelmitta vuorokauden ympäri, vuoden jokaisena päivänä, teknisen infrastruktuurin on oltava huolellisesti suunniteltu. Tällä tarkoitetaan sitä, että palvelinkone sijaitsee puhtaassa ja ilmastoidussa tilassa. Valitsimme projektin palvelintilaksi Laurean Leppävaaran toimipisteessä sijaitsevan palvelinhuoneen.

Yhteyden palvelinkoneeseen olemme ottaneet etänä ssh:n (secure shell) eli salattuun yhteyteen tarkoitettu protokolla tai vnc-ohjelmistoa hyödyntäen, jolloin olemme voineet konfiguroida esimerkiksi palvelinasetuksia. Vnc-ohjelmistoa hyödyntäen olemme luoneet uusia virtuaalipalvelimia, hallinnoineet ja seuranneet virtuaalikoneiden toimintaa.

Fyysisenkoneen asennuksen ja lopullisen sijoituksen paikan valitsemisen jälkeen aloitimme tarvittavien virtuaalipalvelimien luomisen. Ensimmäiseksi luotiin LAMP-palvelin, jonka tarkoituksena on toimia back end -ympäristössä. LAMP-palvelimeen olemme luoneet tiedoston polkuun /etc/httpd/conf.d/sites.conf, jossa on näennäispalvelimen konfiguraatiotiedostot.

Tämän jälkeen loimme virtual manager-ohjelmistolla front end -välityspalvelimen. Front end -välityspalvelimeen teimme tiedoston polkuun /etc/httpd/conf.d/proxy.conf, johon lisäsimme säännön, jonka avulla välityspalvelimeen tulevat pyynnöt välitetään back end -ympäristöön, jossa sijaitsevat nettisivut.

6.1 Suunnittelu

Projektin suunnittelu aloitettiin toimeksiantajan tarpeiden kartoittamisella. Alussa selvitettiin tiomeksiantajalta, minkälaiseen käyttöön virtuaalinen palvelinympäristö tulisi rakentaa. Kokonaiskuvan hahmottumisen jälkeen keskustelimme projektin ohjaajan kanssa projektin hyväksymisestä ja sen alustavasta rajauksesta. Kaiken tämän jälkeen teimme alustavan aikataulun sekä kartoituksen tarvittavista laitteista ja arvion niiden kustannuksista.

Kun laitekustannukset ja vaatimukset saatiin selvitettyä, aloimme tutkia eri yritysten virtuaalisia palvelinratkaisuja. Heti alussa kävi selväksi, että soveltaisimme Amazonin pilvipalveluratkaisuja. Eri palvelinohjelmisto ratkaisuissa nousi esille kolme varteenotettavaa vaihtoehtoa; varnish-, nginx- ja apache-palvelinohjelmistot.

6.2 Fyysinen palvelinkone

Toimeksianto on vaatinut palvelintietokoneen, jonka on pystyttävä paravirtualisoimaan virtuaalisia palvelimia. Tarvittavan koneen olemme saaneet Leppävaaran Laurean Red Labs:lta. Kyseinen laite on HP:n valmistama palvelinkone mallia ”HP ProLiant ML310 G5”, jossa on riittävästi tehoja ajamaan useampia virtuaalipalvelimia samanaikaisesti (HP-palvelinkoneen tekniset tiedot 2011).

Jouduimme kuitenkin tekemään joitakin muutoksia palvelinkoneeseen. Alussa palvelinkoneessa oli kaksi gigatavua muistia. Lisäsimme muistia kahdeksaan gigatavuun asti, koska arvioimme alkuperäisen muistikapasiteetin olevan liian pieni virtuaalisen palvelinympäristön pyörittämiseen. Palvelinkoneessa on myös kaksi verkkokorttia, joka on tärkeätä, kun halutaan kytkeä kone useaan eri verkkoon samanaikaisesti.

6.3 Käyttöjärjestelmä

Palvelinkoneen käyttöjärjestelmänä toimii CentOS Linux-palvelinkäyttöjärjestelmä. Käyttöjärjestelmä perustuu Redhat Enterprise Linux-käyttöjärjestelmään, joka on hyvin suosittu yrityskäytössä. Redhatia käytetään mm. Amazon EC2 Pilvipalveluissa, joka on yksi suurimmista pilvipalveluiden toimittajista. Centos siis perustuu täysin Redhatin lähdekoodiin ja ainoana erona on Centosista on poistetut Redhat yritykseen liittyvät bärndimerkinnät kuten logot ja graafinen ulkoasu. Muita käyttöjärjestelmän eroja ovat Redhatin maksullinen tuki sekä tuoreimmat tietoturva päivitykset ja lisenssimaksut. (Centos 2011)

6.4 Testaus

Valitsimme testiympäristöksi tl-laboratorion. Testiympäristön tarkoituksena oli tutkia eri ratkaisuvaihtoehtoja ja kerätä tietoa niiden toimivuudesta. Tavoitteena oli, että testiympäristössä havaittujen tietojen pohjalta lähdettäisiin rakentamaan käytännön virtuaaliympäristöä, jossa opiskelijat pystyisivät ajamaan nettisivujaan.

Testi koneena käytettiin virtualisoitua front end palvelinta, joka on osa virtuaalipalvelin ympäristöä. Koneen valinta persustuu kyseisen koneen loogisella sijainnilla verkossa sekä sen

tehtävän perusteella. Front end palvelin toimii välitys palvelimena ja käsittelee ulkoverkosta tulevat pyynnöt joten sopivan ja varmatoimisen palvelin ohjelmiston löytäminen oli tärkeää.

6.4.1 Varnish-cache

Varnish-cache on avoimeen lähdekoodiin perustuva välityspalvelin, jota käytetään pääsääntöisesti Web-palvelimien nopeuttamiseen. Alustava ohjelmisto, jota testasimme oli varnish-cache välimuisti, jota voi käyttää myös välityspalvelimena. Palvelin toimi nopeasti ja välitti verkon ulkopuolelta tulevat pyynnöt back end -ympäristöön ja takaisin. Varnish-cachen ominaisuuksiin kuului myös välimuisti, jonka tarkoituksena on nopeuttaa pyyntöjen käsittelyä ja vähentää verkkoliikennettä back end -ympäristössä. Testien aikana kuitenkin huomasimme, että kyseinen palvelinohjelmisto ei pysty käsittelemään https-liikennettä. Ratkaisuna ongelmaan oli käyttää ”Pond proxy” nimistä ohjelmistoa, joka purkisi ssl (secure sockets layer) salauksen https-paketeista ja lähettäisi ne edelleen varnish-cache välityspalvelimelle. Testauksen aikana päätimme kuitenkin hylkää kyseisen teknisenratkaisun monimutkaisuuden vuoksi.(Varnish-cache palvelinohjelmisto. 2012.)

6.4.2 Nginx

Nginx on avoimeen lähdekoodiin perustuva http-palvelinohjelmisto, jota voidaan käyttää myös välityspalvelimena (reverse proxy). Nginx on suhteellisen uusi http-palvelin tulokas. Nginx:n kehittäminen aloitettiin vuonna 2002 ja ensimmäinen versio julkaistiin vuonna 2004. Nginx on tunnettu sen nopeudesta vakaudesta sekä monipuolisista ominaisuuksista.(Nginx wiki)

Testauksen aika kuitenkin törmäsimme ongelmaan, joka liityi palvelimen konfiguraatio tiedostojen muokkaamiseen. Nginx sivulla mainittiin, että palvelimen konfiguroiminen toimivaan kuntoon on helppoa. Huomasimme, että käytännössä apachen konfuguroiminen sekä muutosten tekeminen oli paljon helpompaa. Päätimme hylkää kyseisen palvelin ohjelmiston siihen kuluvan työnajan takia. Lisäksi Nginx ei kuulu Centosin pakettivalikoimaan ja siksi sen päivittäminen olisi jouduttu tekemään manuaalisesti, mikä vaikeuttaa palvelimen ylläpitoa ja luo tietoturvariskejä.

6.4.3 Apache

Apache on suosittu avoimeen lähdekoodiin perustuva http/web-palvelinohjelma. Apachen tunnetuin tuote on httpd-palvelin, joka on markkinajohtaja reilun 60 prosentin markkinaosuudellaan. Microsoft IIS (Internet information server) jää toiseksi 30 prosentin osuudellaan.

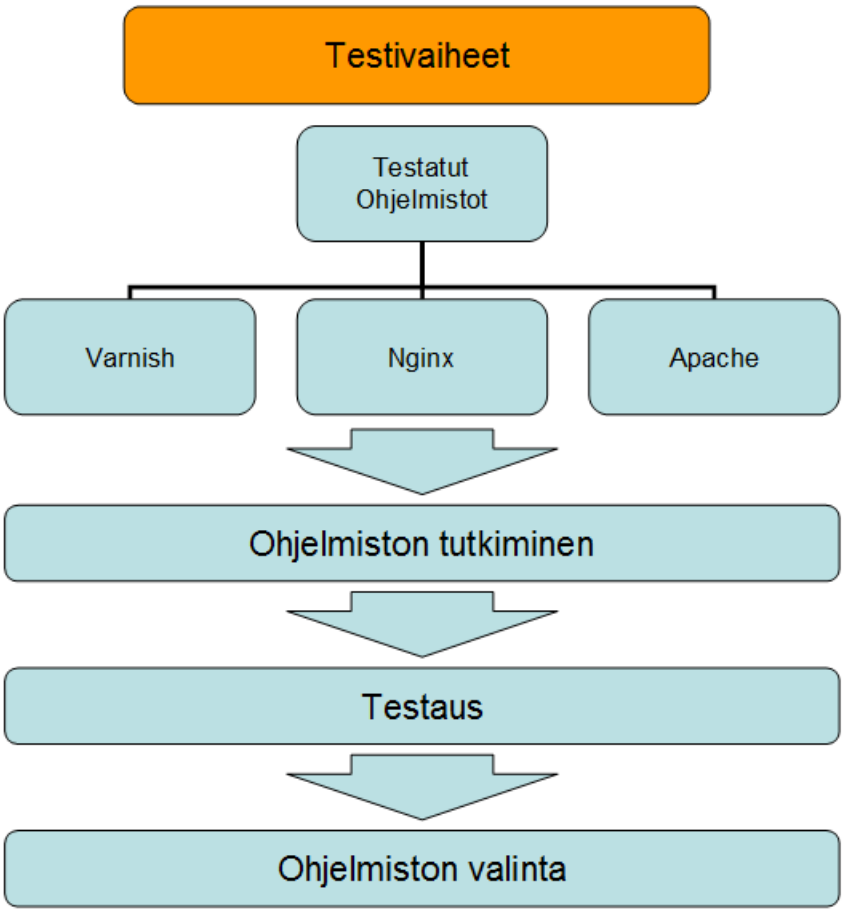
Apachen historia alkaa vuodesta 1995, jolloin monen eri tahojen kehittäjät halusivat luoda paremman httpd-palvelinjärjestelmän. Eri kehittäjäjoukot yhdistyivät, minkä seurauksena perustettiin Apatche server ja Apache group.

Apachen menestyksen taustalla on useita eri tekijöitä. Maineeltaan Apache on erittäin nopea, varma ja vakaa palvelinohjelma. Avoimen lähdekoodin ansiosta Apache on saanut jalansijan useissa kaupallisissa palvelinratkaisuissa. Lisäksi Apache on saatavilla myös muille alustoille, kuten Windowsille ja Amigalle. (Apache-koulutus 2011)

Apache-ohjelmisto oli lopullinen valinta palvelinympäristöön. Valintaan vaikutti ohjelmiston ominaisuuksiin kuuluvat välityspalvelun (reverse proxy) välimuistiominaisuudet sekä https-protokolla.

6.4.4 Välityspalvelimen testaus

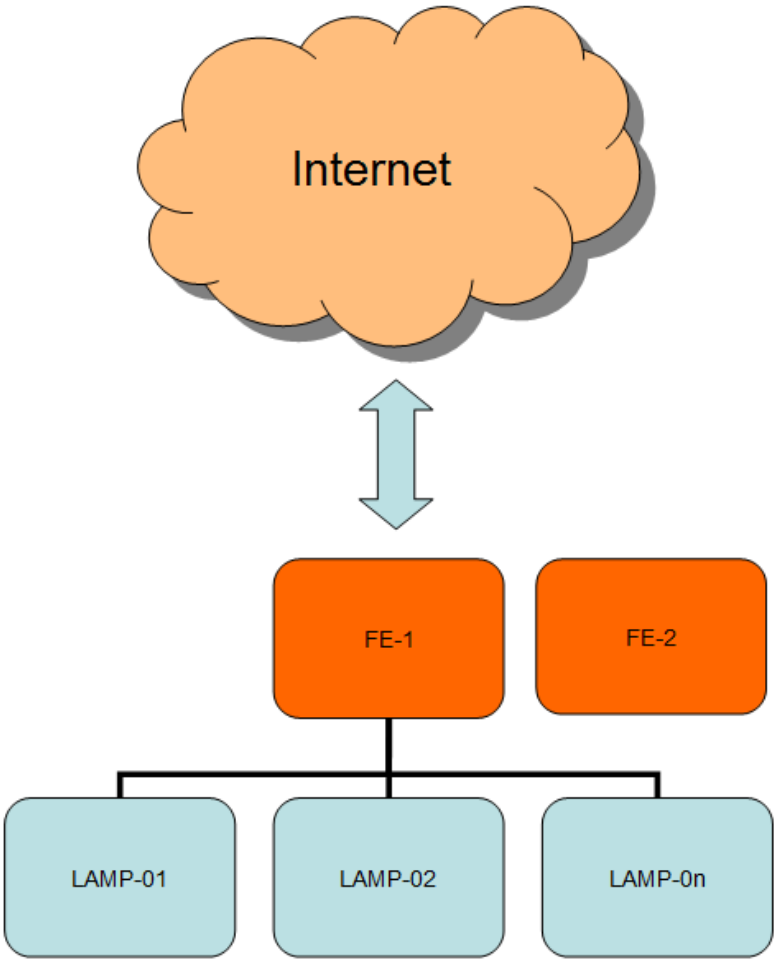
Lopullinen testaus toteutettiin siten, että ajoimme Apachen testisivut back end -ympäristössä sijaitsevassa lamp-palvelimessa (Linux, Apache, MySQL ja PHP). Onnistuneen testivaiheen jälkeen olemme siirtyneet ajamaan opiskelijoiden nettisivuja virtuaalisessa palvelinympäristössä. Seuraavassa kuviossa neljä on kuvattu prosessi testauksesta ohjelmiston valintavaiheeseen.



Kuvio 4: Testivaiheen eteneminen

6.5 Käyttöönotto

Virtuaaliympäristön toteutus saatiin vuoden 2011 talven aikana siihen vaiheeseen, että opiskelijat ovat päässeet laittamaan nettisivunsa back end -palvelinympäristöön. Projektin tekijät ylläpitävät ja vastaavat palvelinympäristön toimivuudesta. Mahdollisesta jatkosta neuvotellaan toimeksiantajan kanssa. Tarkoituksena on ollut, että ylläpitotehtävät siirtyvät tl-laboratorion harjoittelijoiden vastuulle. Seuraavassa kuviossa X näkyy tämän hetkisen virtuaalisen palveliympäristön rakenne. Seuraavan kuvion viisi lyhenne FE-1 tarkoittaa front end –välityspalvelinta, joka käsittelee ulkoverkosta tulevia http-pyyntöjä. FE-2 on kopio FE-1:stä, mutta toimii reservipalvelimena. Välityspalvelinten alla olevat LAMP-palvelimet sisältävät niin sanotut LAMP-ohjelmisto pinon, joka mahdollistaa web-sovellusten ajamisen.



Kuvio 5: Virtuaalinen palvelinympäristö

7 Ylläpidolliset tehtävät

Ylläpito kuuluu monipuoliseen ja laaja-alaiseen tietohallinnon osa-alueeseen. Ylläpitotehtävät vaihtelevat sen mukaan mitä järjestelmiä ja ohjelmistoja on sekä mihin niitä käytetään, missä roolissa ylläpitäjä organisaatiossa toimii ja kuinka monta ylläpitäjää on. Ylläpitotehtävät voidaan jakaa kolmeen pääosaan: ohjelmiston, laitteiston sekä sisällön ja toiminnallisuuden ylläpidollisiin tehtäviin. Ylläpitäjän parhaita oppaita ovat niin sanotut ”best practise” sivustot, joissa käydään hyvin läpi erilaisia ongelmatapauksia. Näitä sivustoja ovat muun muassa www.xen.org.

Tämän projektin varsinaiset ylläpidolliset tehtävät ovat liittyneet ohjelmiston ylläpitoon. Ylläpidon viikoittaisiin rutiineihin kuuluvat mm. lokitiedostojen tarkastaminen, kaatuneiden prosessien uudelleen käynnistäminen, ongelmatilanteiden ratkaiseminen ja tietoturvasta huolehtiminen. Edellä mainituista tehtävistä tietoturvasta huolehtiminen on ehkä tärkein etenkin laitteilla, jotka ovat suoraan yhteydessä avoimeen verkkoon. Tässä projektissa tällaisia palvelinkoneita ovat front end -välityspalvelin, jonka lokitiedostojen päivittäinen

tarkastaminen on ensisijaisen tärkeää. Tämä johtuu siitä, että kyseinen laite on suoraan yhteydessä ulkoverkkoon ja käsittelee sieltä tulevat pyynnöt porteissa 80, 443 ja 2202. Projektin testauksen yhteydessä on ollut havaittavissa useita hyökkäysyrityksiä välityspalvelimen ssh-palveluun sekä http-palveluun, joka toimii välityspalveluna Back end -palvelinympäristölle.

Alla olevasta esimerkistä näkyy ”ZeMu” hyökkäys, joka on kohdistettu back end -palvelimella toiminnassa olevaan PhPmyadmin hallintatyökalun php-skripteihin.

```
186.46.43.18 - - [18/Nov/2011:08:42:50 +0200] "GET /admin/scripts/setup.php HTTP/1.1" 404 1138 "-" "ZmEu"
186.46.43.18 - - [18/Nov/2011:08:42:52 +0200] "GET /admin/pma/scripts/setup.php HTTP/1.1" 404 1138 "-" "ZmEu"
186.46.43.18 - - [18/Nov/2011:08:42:53 +0200] "GET /admin/phpmyadmin/scripts/setup.php HTTP/1.1" 404 1138 "-" "ZmEu"
186.46.43.18 - - [18/Nov/2011:08:42:54 +0200] "GET /db/scripts/setup.php HTTP/1.1" 404 1138 "-" "ZmEu"
186.46.43.18 - - [18/Nov/2011:08:42:55 +0200] "GET /dbadmin/scripts/setup.php HTTP/1.1" 404 1138 "-" "ZmEu"
```

7.1 Esimerkkejä ylläpidollisista tehtävistä

7.1.1 Case 1: Lamp01-palvelimen verkko ongelma

”Yrittäessämme ottaa yhteyttä lamp01-palvelimeen, huomasimme että siihen ei saanut yhteyttä ping-komennolla. Kirjauduimme palvelimeen konsoli yhteydellä komennolla ”xm console lamp01”. Tarkistimme komennolla ”ifconfig” verkkokortin tilan, joka oli kunnossa. Komennolla ”route” tarkistimme myös reititystaulun. Lisäksi käynnistimme komennolla ”service network restart” verkkopalvelun uudestaan. Nämäkään toimenpiteet eivät auttaneet. Lopulta päätimme käynnistää koko lamp01-palvelimen uudestaan komennolla ”reboot”. Tämä auttoi verkon ongelmiin, mutta jouduimme erikseen käynnistämään mysqld- ja httpd-palvelut.”

Httpd ja mysqld palvelujen manuaalisen käynnistyksen välttämiseksi olemme lisänneet ne automaattisesti käynnistyvien palveluiden listaan chkconfig-komennolla. Chkconfig komennon avulla on mahdollista lisätä haluttu palvelu käynnistymään palvelimen käynnistyksen yhteydessä. Komento syntaksi on seuraava ”chkconfig –level 3 mysqld on”. Kyseinen komento esimerkki lisää mysqld palvelun käynnistettävien palveluiden listaan. Chkconfig --list

komennolla voi myös tarkistaa kaikkien palvelujen ja ajotasojen tilan ja muuttaa niitä tarpeen mukaan.

7.1.2 Case 2: Virt-manager ohjelma ei luo uusia virtuaali palvelimia

Yrittäessämme luoda uutta virtuaalista palvelinta virt manager teki virhe ilmoituksen. Käynnistimme libvirtd- ja xend-palvelut ”service” komennolla, jonka jälkeen ongelma hävisi. libvirtd on virtuaalisen järjestelmän hallintakomponentti. Se on ajossa hallintakoneessa ja suorittaa tarvittavia hallintatehtäviä virtualisoiduille vieraskoneille. Sillä voidaan muun muassa käynnistää, sammuttaa, siirtää pyyntöjä palvelimilta toisille, ohjata verkkoliikennettä ja hallita pyyntöjen varastointia.

8 Palvelinympäristön hallinnointiohjelmistot

Palvelinympäristön toteuttaminen aloitettiin määrittelemällä sen käyttötarkoitus. Tässä projektissa tarkoituksena on ollut virtualisoida palvelimia nettisivujen ylläpitoon. Määrittelyn jälkeen olemme kartoittaneet palvelinkäyttöjärjestelmiä, jotka sopisivat parhaiten nettisivujen ajamiseen.

Järjestelmän asennuksen aikana valitsimme minimimäärän ohjelmistoja, koska emme halunneet että ylimääräiset ohjelmistot kuluttaisi liikaa palvelinkoneen resursseja. Valitut ohjelmistot ovat pääsääntöisesti välttämättömiä virtualisointi tuki- ja hallintatyökalu-ohjelmistoja. Alla on luettelo kyseisistä ohjelmistoista, jotka asennettiin fyysiseen palvelinlaitteeseen:

-Virtual machine manager
-VNCserver
-Virt clone
-Webmin-hallintapaneeli
-Xen hypervisoriin liittyvät hallintatyökalut (xm-komento)

Edellä mainituilla ohjelmistoilla on ollut suuri merkitys palvelinympäristön toteuttamisessa ja ylläpidossa. Esimerkiksi hallintatyökaluilla, kuten Virtual machine manager ohjelmistolla muokataan virtuaalisia palvelimia. Kyseisellä ohjelmistolla voidaan myös seurata paljon kukin virtuaalikone kuluttaa suorittimen muistia(katso Liite 1, kuvio 4: Uuden virtuaalikoneen luominen). Muistin kuormituksen seuraaminen on tärkeätä, jotta voidaan estää virtualisen palvelimen ylikuormittuminen tai pahimmassa tapauksessa sen kaatuminen.

Vnc-server on etähallintaan käytettävä työkalu. Tässä projektissa olemme käyttäneet vnc-server ohjelmistoa järjestelmän hallinnan helpottamiseksi. Asensimme vnc-serverin ”Domain0” hallintakoneelle, jolla pystytään hallinoimaan virtuaalista palvelinympäristöä. Vnc-server on varsin helppokäyttöinen ja on hyödyllinen mikäli ei ole aikaisempaa kokemusta terminaalin käytöstä. Vnc-serverin käyttö tapahtuu käytännössä siten, että etäkoneella otetaan yhteys vncserver-ohjelmistolla Domain0 hallintakoneeseen.

Virt clone-ohjelmisto on nimensä mukaisesti kloonausohjelmisto. Sen avulla voidaan luoda kopio jo olemassa olevasta virtuaali palvelimesta. Tämä on hyödyllistä, jos halutaan ottaa nopeasti käyttöön esimerkiksi lamp-palvelin samoilla asetuksilla. Tällöin ei tarvitse käydä koko asennusprosessia läpi. Virt-clone olemme käyttäneet testausvaiheessa. Kyseistä ohjelmistoa voidaan käyttää terminaalissa alla olevilla komennoilla:

```
virt-clone -o FE -n FE-1 -f /var/lib/xen/images/uudenkoneenkovalevyimage.img
```

Alla komennossa käytettävät parametrit:

-o: alkuperäisen virtuaalikoneen nimi.
-n: uuden virtualisen koneen nimi.
-f: polku jossa uuden koneen levykuva tiedosto tulee sijaitsemaan kopioinnin jälkeen.
-d: Debug toiminto jonka avulla voidaan diagnosoida mahdollisia vikatilanteita kopiointi prosessin aikana.

Webmin on hallintapaneeli, joka on suunniteltu Unix-pohjaisiin käyttöjärjestelmiin. Webminia käytetään selaimessa ja sillä voidaan muokata muun muassa Lamp-palvelimen käyttäjiä, DNS-tietueita ja tiedostojen jakoa. Webmin on hyödyllinen ohjelmisto ylläpitäjälle, sillä pystyy hallinnoimaan kerralla kaikkia virtualikoneissa olevia nettisivuja.

Xen-hypervisoriin kuuluvilla hallintatyökaluilla tarkoitetaan niin sanottua xm-komentoa terminaalissa, jolla voidaan käynnistää, sammuttaa, luoda yhteyksiä tai listata virtualikoneita. Xm-komennoilla voidaan nopeasti vaikuttaa virtualikoneisiin, yksittäiseen nettisivustoon tai tietokantaan. Xen-hypervisor työkalujen käyttäminen vaatii kokemusta, jonka vuoksi aloittelevan ylläpitäjän olisi hyvä tutustua xm-komentoihin. Virtuaalikoneita voidaan hallita muun muassa seuraavanlaisilla xm-komennoilla:

xm create /etc/xen/lamp01	Käynnistää lamp01 nimisen virtualikoneen
xm shutdown lamp01	Sammuttaa lamp01 virtualikoneen
xm list	Listaa käynnissä olevat virtualikoneet
xm help	Listaa kaikki xm-komennot

9 Tietoturva

Tietoturvalla tarkoitetaan yleisesti tietojen, palveluiden ja tietoliikenteen suojaamista. Tietoturva rakentuu sanan ”tiedon” kolmen ominaisuuden mukaan; luottamuksellisuus, eheys ja käytettävyys. (Viestintävirasto. 2012).

Tässä projektissa tietoturva voidaan jakaa kolmeen pääkohtaan; Verkon, palveluiden ja yhteyksien tietoturvaan. Jokaisesta kohdasta on tehty oma kappale, jossa kerrotaan tarkemmin mitä tietoturvaratkaisuja projektissa on tehty kyseisillä osa-alueilla.

9.1 Verkon tietoturva

Virtuaalinen palvelinympäristö toimii tl-laboratorion verkossa ja sen tietoliikenne on eristetty Laurean muusta verkosta. Lisäksi palvelinympäristö on suojattu tl-laboratorion ASA palomuurilla. Palvelinympäristössä on myös yksi välityspalvelimena toimiva virtuaalinen kone, jota kutsutaan front end -palvelimeksi. Kyseisen palvelimen tarkoituksena on sallia http- ja https-liikenteet ulkoverkosta tl-labolatorion verkkoon, jossa sijaitsee back end -ympäristön lamp-palvelinkoneet.

9.2 Palvelujen tietoturva

Palvelinympäristössä on yksi virtuaalinen palvelinkone, jonka tarkoituksena on toimia välityspalvelimena ulkoverkon ja back end -ympäristön välissä. Tätä virtualisoitua palvelikonetta kutsutaan front end -palvelimeksi. Kyseisessä palvelimessa on muista virtualisista palvelimista poiketen kaksi verkkokorttia, joista toisella on ulkoinen ip-osoite ja toisella sisäinen ip-osoite. Kyseisen ratkaisun tarkoituksena on mahdollistaa pääsy Laurean verkon ulkopuolelta käyttäjien kotisivuille. Koska kyseinen laite näkyy Internetissä, on sen turvaaminen elintärkeää muiden palvelinympäristössä toimivien palvelimien kannalta. Tällä hetkellä front end -palvelinkoneella toimivat apache- ja ssh-palvelut. Apache-palvelimen tarkoituksena on tässä palvelinkoneessa toimia välityspalveluna, joka ottaa vastaan verkon ulkopuolelta tulevat http-pyynnöt ja lähettää ne edelleen back end -ympäristön palvelimille, josta back end -ympäristön palvelin palauttaa pyynnöt takaisin.
Kuten muissa virtuaalisissa palvelinympäristön koneissa, myös front end:ssa on ssh-palvelu, jota käytetään palvelinkoneiden etähallintaan. Koska ssh:n avulla voi hallita koko palvelin ympäristöä, olemme rajoittaneet ssh-yhteydet vaihtamalla porttiin, joka ei näy Laurean verkon ulkopuolella. Lisäksi olemme estäneet kirjautumisen pääkäyttäjänä kaikista virtuaalikoneiden ssh:n palvelun asetuksista. Tämä tietoturvaratkaisu perustuu siihen, että kaikissa Unix-pohjaisissa käyttöjärjestelmissä pääkäyttäjän nimi on ”root” ja koska tämä on yleistä tietoa, murtautujat yrittävät aina ensiksi arvata ”root” käyttäjän salasanaa. Estämällä

”root” käyttäjän kirjautumisen, hyökkääjä joutuu arvaamaan sekä tavallisen käyttäjän käyttäjätunnuksen ja sen salasanan, mikä vaikeuttaa huomattavasti murtautumista palvelinympäristöön. Toinen tapa jolla murtautumisyrityksiä voi rajoittaa, on luoda käyttäjäryhmä ja lisätä siihen käyttäjät, joilla on lupa kirjautua ssh-palveluun. Tämä toimenpide evää pääsyn kaikilta muilta käyttäjiltä, jotka eivät ole kyseisessä ryhmässä.

9.3 Yhteyksien rajoittaminen

Front end järjestelmän testauksen aikana huomasimme http-lokitiedoista, että palvelin saa epämääräisiä pyyntöjä ja muuta liikennettä ulkomailta. Jäljitimme IP-osoitteet, joista pyynnöt tulivat ja selvisi, että osa pyynnöistä tuli Itä-Euroopasta ja Aasiasta. Päätimme estää kaiken liikenteen Suomen palveluntarjoajien ulkopuolelta lisätäksemme järjestelmien ja tl-labolatorion verkon tietoturvaa.

9.4 Loppukäyttäjien tietoturva

Tässä projektissa loppukäyttäjien tietoturva on jätetty harjoittelijoiden ja opiskelijoiden omalle vastuulle. Tällä tarkoitetaan sitä, että jokainen joka laittaa nettisivunsa back end -ympäristöön, vastaa itse oman koodinsa tietoturvasta. Esimerkiksi MySQL-tietokannan tietoturva on jätetty käyttäjien vastuulle. MySQL-tietokannan tietoturvaa voidaan helposti parantaa tietoturvallisilla käyttäjätunnuksilla ja salasanoilla, jotka käyttäjä itse luo MySQL-tietokantaan.

10 Projektin ongelmakohdat

Projektin suurimmat ongelmakohdat ovat olleet ohjelmistoihin ja arkkitehtuurisiin ratkaisuihin liittyvät valinnat. Paras toteutus on saavutettu suunnittelulla ja testauksella. Tämän takia, projektissa on yritetty selvittää niin sanottua "best practice" eli paras mahdollinen palvelinratkaisu menetelmä, jolla pyritään saavuttamaan mahdollisimman vakaa ja tehokas virtuaalinen palvelinympäristö. Oikean ohjelmiston valitseminen on ollut tärkeätä, sillä ohjelmiston on pitänyt olla yhteensopiva laitteiston kanssa.

Arkkitehtuurisena ongelmana pidimme alussa muistin määrä, koska se on olennainen osa virtuaalisten palvelinten toiminnassa. Tämä ongelma selvitettiin lisäämällä palvelinkoneeseen muistia. Yleisesti palvelinvirtuaalisoinnin suurin ongelma ei ole keskusyksikön suorituskyky, vaan palvelinpään liian pieni muistin määrä. Lisäksi virtuaalisten palvelinten tietoturva voi koitua kompastuskiveksi, mikäli siihen ei suunnitelmavaiheessa kiinnitetä huomiota.

11 Projektin toteutus, eteneminen ja aikataulu

Projekti käynnistyi vuoden 2011 helmikuussa. Projekti aloitettiin orientointivaiheella, jossa ideoitiin ja keskusteltiin toimeksiantajan kanssa. Keskustelussa kartoitettiin erityisesti toimeksiantajan toiveita ja tarpeita projektille. Kun asioista päästiin yhteisymmärrykseen, aloitimme varsinaisen tutkimussuunnitelman työstämisen.

Tutkimussuunnitelmassa pohdimme alussa molempien tekijöiden aikataulujen yhteen sovittamista ja sen jälkeen sovimme viikonpäivät, jolloin työskentelisimme projektin parissa. Suunnitteluvaiheessa keskityimme erityisesti hahmottamaan ja ymmärtämään xen-virtualiympäristön kokonaisuuden. Tämän jälkeen teimme tutkimustyötä, jossa selvitimme muun muassa mikä olisi sopivin ratkaisu palvelinkoneen käyttöjärjestelmäksi ja mitä erityisiä laitevaatimuksia palvelinkone vaatii. Koska pyrimme säästämään kustannuksista, valitsimme Linuxin CentOS-palvelinkäyttöjärjestelmän. Valinta oli lopulta helppo, koska tekijöillä oli kokemusta kyseisestä palvelinkäyttöjärjestelmästä ja he olivat todenneet sen toimivaksi. Lisäksi kyseisen käyttöjärjestelmän yritysversiota Red Hatia käytetään myös monissa yrityksissä.

Projektin aikana kokoonnuimme noin kolme kertaa viikossa. Aikataulullisia muutoksia on tapahtunut muun muassa tekijöiden työharjoittelujaksojen takia. Projekti saatiin päätökseen suunnitelman mukaan joulukuussa 2011.

12 Pilvipalvelut vaihtoehtona?

Virtualisoinnin yhteydessä puhutaan usein myös pilvipalveluista, sillä pilvipalvelussa käyttäjät eivät voi nähdä tietyn Internet palvelun teknisiä yksityiskohtia. Ideana pilvipalvelussa on, että Internet palveluun saadaan resursseja niin sanotusti ”pilvestä” olevissa palvelinkoneista tai niiden palveluista. Pilvipalvelut helpottavat huomattavasti muun muassa yksittäisten palvelinten kuormantasoa. Projektin aikana on pohdittu jatkokehityksenä pilvipalvelun toteuttamista virtuaalisessa palvelinympäristössä.

Pilvipalvelumallit voidaan esittää viiden kerroksen pinoina asiakas-sovellus-alusta-infrastruktuuri-palvelin (client-application-platform-infrastructure server). Edellä mainittujen kerroksien mukaan pilvipalvelut voidaan luokitella kolmeen perustyyppiin: palvelualustan ulkoistaminen (Platform as a service) eli PaaS, palvelimien ja palvelinsalien ulkoistaminen (Infrastructure as a service) eli IaaS ja ohjelmiston hankkimista palveluna (Software as a Service) eli SaaS . Näiden pilvipalveluiden yhdistelmää kutsutaan nimellä hybridipilvi (Hybrid cloud) eli yhdistelmä useista pilvipalveluista.(Heino 2010, 50) Projektin jatkon kannalta PaaS

ja IaaS pilvipalvelumalleja olisi mahdollista soveltaa virtuaalisessa palvelinympäristössä. Alla selvennetään miten kyseiset pilvipalvelut voisivat toimia tl-laboratoriossa.

PaaS-pilvipalvelu voitaisiin toteuttaa tl-laborion palvelinympäristössä siten, että opiskelija pystyisi selaimella ottamaan yhteyden palvelinympäristössä toimivaan palveluun sovellus-rajapinnan välityksellä ja pääsisi näin käsiksi palvelinympäristön virtuaalikoneeseen asennettuihin sovelluksiin. Sovellus voisi olla esimerkiksi Laurean käyttämä Optima sovellusalusta. Opiskelija ei pääsisi käyttämään kuin määritettyä sovellusta virtuaalikoneessa. PaaS-pilvipalvelu tuo omat haasteensa, sillä ylläpitäjinä tl-laboratorion harjoittelijoiden täytyisi huolehtia sovellusalustan tuotantoympäristön pystyttämisestä ja ylläpidosta. (Heino 2010, 51)

IaaS-pilvipalvelua voisi soveltaa tl-laboratoriossa siten, että tl-laboratorio olisi palveluntarjoaja. Palveluntarjoajana se lohkoisi palvelinympäristöstä virtuaalikoneita oppilaiden käyttöön. Erona PaaS-pilvipalveluun, IaaS-pilvipalvelussa oppilaat pääsisivät käsiksi virtuaalikoneen omaan näennäispalvelimeen, johon he voisivat asentaa esimerkiksi Drupal-sisällönhallintajärjestelmän. Kyseisiä näennäispalvelimia voitaisiin asentaa virtuaalikoneeseen useampia. Näennäispalvelimet voisivat olla tässä tapauksessa esimerkiksi webhotelleja, joissa olisi kotisivutilaa ja sähköpostipalveluita. Webhotelleja voitaisiin jakaa oppilaille tarpeen mukaan eri hankkeisiin. (Heino 2010, 52-53)

13 Kehitysehdotus

Projektin jatkon kannalta on tärkeää, että virtuaalisen palvelinympäristön toimintaa kehitetään. Tl-laboratorion harjoittelijat ovat ensisijaisessa roolissa kehitystyössä. Harjoittelijat pystyvät parhaiten havainnoimaan kehitettäviä kohteita sekä ovat ajantasalla palvelimen käyttöasteesta. Toimeksiantajan halukkuudella on myös vaikutusta jatkokehitykselle. Suurimmat kehityskohteet liittyvät ohjelmistoihin ja laitteistoihin.

Tällä hetkellä HP proliant-palvelinkoneella pystyy paravirtualisoimaan noin kahdeksaa virtualisoitua konetta. Tehokkaamman isäntäkoneen hankkiminen on tarpeellista, mikäli se otetaan aktiiviseen opetuskäyttöön tai halutaan lisää virtuaalisia palvelinkoneita. Front end -palvelin on tällä hetkellä virtuaalinen kone, joka käsittelee pyynnöt ja lähettää ne edelleen back end -ympäristöön. Fyysinen front end -palvelin, jossa olisi kaksi verkkokorttia voisi olla nykyistä nopeampi, vakaampi ja turvallisempi vaihtoehto.

Varsinainen mahdollinen jatkoprojekti olisi rakentaa palvelinympäristölle varmuuskopiointi. Varmuuskopiointi on yksi olellisimmista ja tärkeimmistä tiedonvarmistukseen käytettävistä menetelmistä. Varmuuskopiointi voitaisiin toteuttaa ulkoiselle kovalevylle, joka ajastettaisiin

niin sanotulla ”cron”-ajastusohjelmalla tai varmuuskopiot otettaisiin manuaalisesti. Ajastettu ”cron” toimisi sitten, että skripti pysäyttäisi virtuaaliset koneet ja ottaisi niistä kopiot ulkoiselle kovalevylle. Varmuuskopiointi voitaisiin toteuttaa esimerkiksi kerran kuukaudessa sunnuntain ja maanantain välisenä yönä, jonka aikana palvelinympäristön käyttö olisi mahdollisimman vähäistä.

Muita tiedoneheyden varmentamiseen liittyviä toimenpiteitä olisi hankkia UPS-varavirtalähde mahdollisten virranjakelun häiriöiden varalta. Pahimmillaan virranjakelutkatkokset voivat johtaa tiedostojen ja virtuaalistenkoneiden tuhoutumiseen tai fyysisen laitteen rikkoutumiseen.

14 Yhteenveto

Projektin tarkoituksena oli toteuttaa virtuaalinen palvelinympäristö. CentOS-käyttöjärjestelmän valitseminen palvelinkoneeseen oli onnistunut valinta. Lisäksi tarkoituksena oli tutustua virtuaalisointiin yleisesti. Aiheena virtualisointi oli laaja ja haastava.

Molemmilla tekijöillä oli jonkin verran kokemusta Linux-käyttöjärjestelmistä, mikä helpotti hieman aiheeseen perehtymistä. Projektin aikana olemme perehtyneet laajalti xen-virtualisointiin sekä virtuaalisen palvelinympäristön kokonaisuuteen.

Projekti oli kokonaisuudessaan onnistunut. Toimeksiantajan toivomuksena oli rakennuttaa virtuaalinen palvelinympäristö, jota voitaisiin hyödyntää tl-laboratoriossa. Rakensimme virtuaalisen palvelinympäristön, jossa voidaan ylläpitää opiskelijoiden nettisivuja. Olemme pyrkineet luomaan mahdollisimman kattavan kokonaisuuden projektin virtuaalisesta palvelinympäristöstä, konfiguraatio tiedostoista, virtuaalikoneen asennuksesta, hallintaohjelmistoista sekä pilvipalveluiden mahdollisesta hyödyntämisestä palvelinympäristössä. Projektin raportti antaa hyvän yleiskatsauksen virtuaalisesta palvelinympäristösä ja xen-virtualisointitekniikasta.

Lähteet

Painetut teokset:

Heino, P. 2010. Pilvipalvelut. Hämeenlinna. Kariston kirjapaino Oy.

Sähköiset lähteet:

Apache-koulutus. 2011.
http://www.2kmediat.com/apache/apachehistoria.asp

Apache välityspalvelin asennuksen ohjeet. 2011.
http://httpd.apache.org/docs/2.2/mod/mod_proxy.html

Centos. 2011.
http://www.centos.org/

Xen virtualisointi. 2011. Citrixs systems corporation.
http://xen.org/

How does xen works. 2011. Citrixs systems corporation.
http://xen.org/files/Marketing/HowDoesXenWork.pdf

Hp-palvelin koneen tekniset tiedot. 2011.
http://h18000.www1.hp.com/products/quickspecs/12858_na/12858_na.HTML

Ip-pohjainen ja portti kohtainen näennäispalvelin asetukset. 2011.
http://httpd.apache.org/docs/2.2/vhosts/examples.html

Laurea Fakta. 2011. viitattu 7.11.2011.
http://www.Laurea.fi/fi/opiskelu/oppaat/Documents/Fakta_final_2011_2012_PB_210611-linkitetty.pdf

Nginx wiki. 2012.
http://wiki.nginx.org/Main

Ohjeita Xen virtualisoimisesta. 2009.
http://www.virtuatopia.com/index.php/Xen_Virtualization_Essentials

Varnish-cache palvelinohjelmisto. 2012.
https://www.varnish-cache.org/

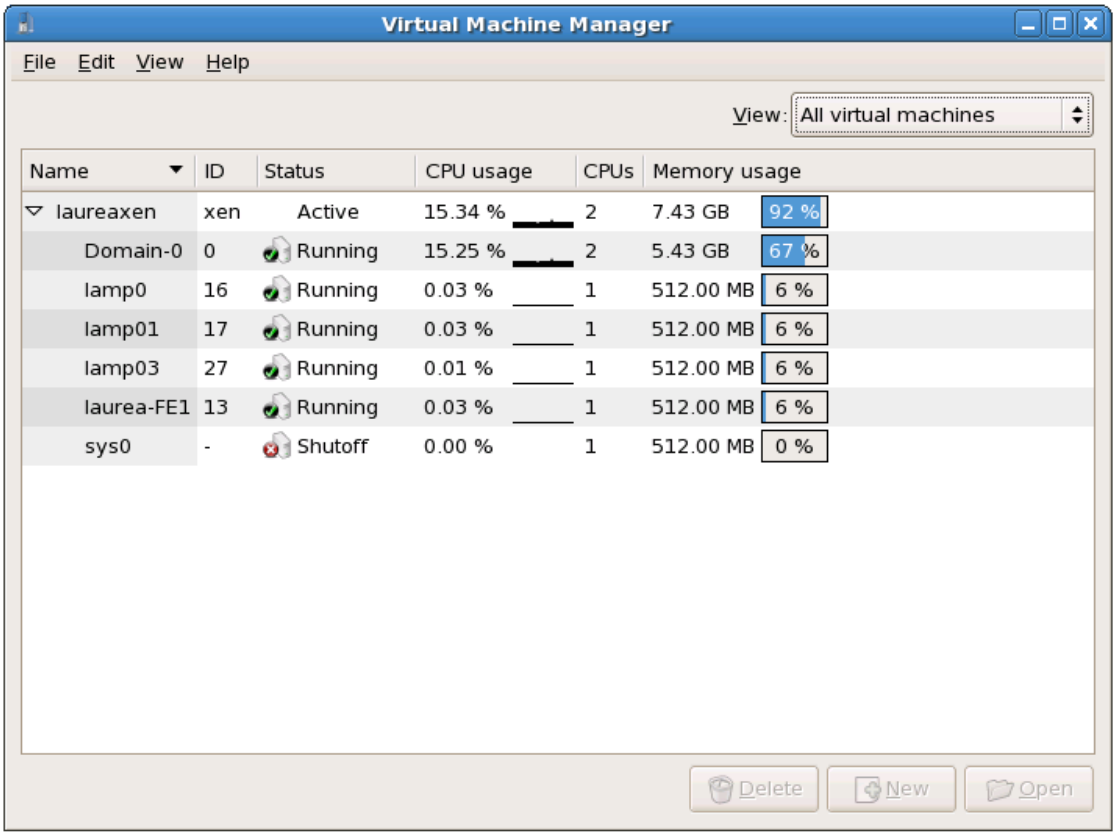
Viestintävirasto. 2012.
http://www.ficora.fi/index/palvelut/palvelutaiheittain/tietoturva.html

Xen-verkkoliitännät. 2011.
http://wiki.xen.org/xenwiki/XenNetworking

Kuviot

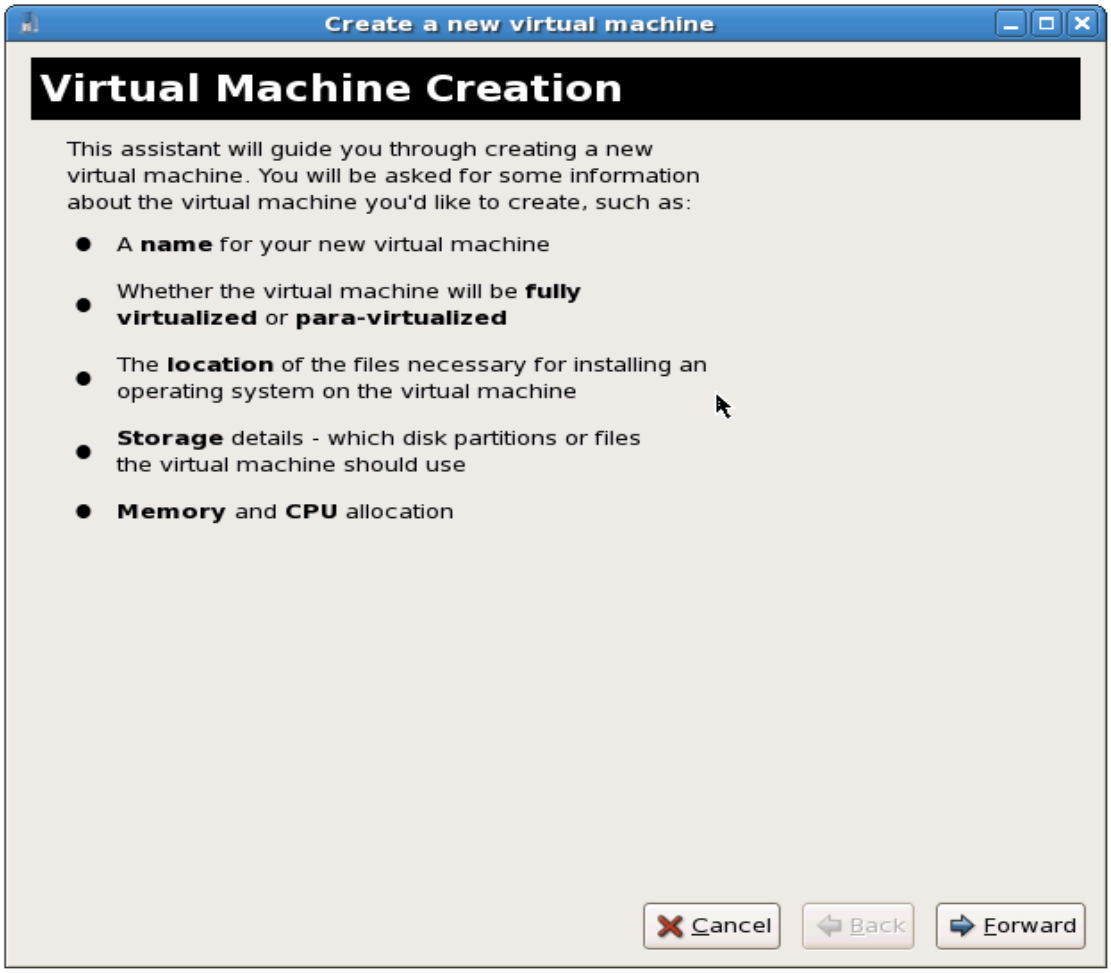
Liitteet

Liite 1 Virtuaalisen palvelimen asentaminen Virtual machine manager-sovelluksella



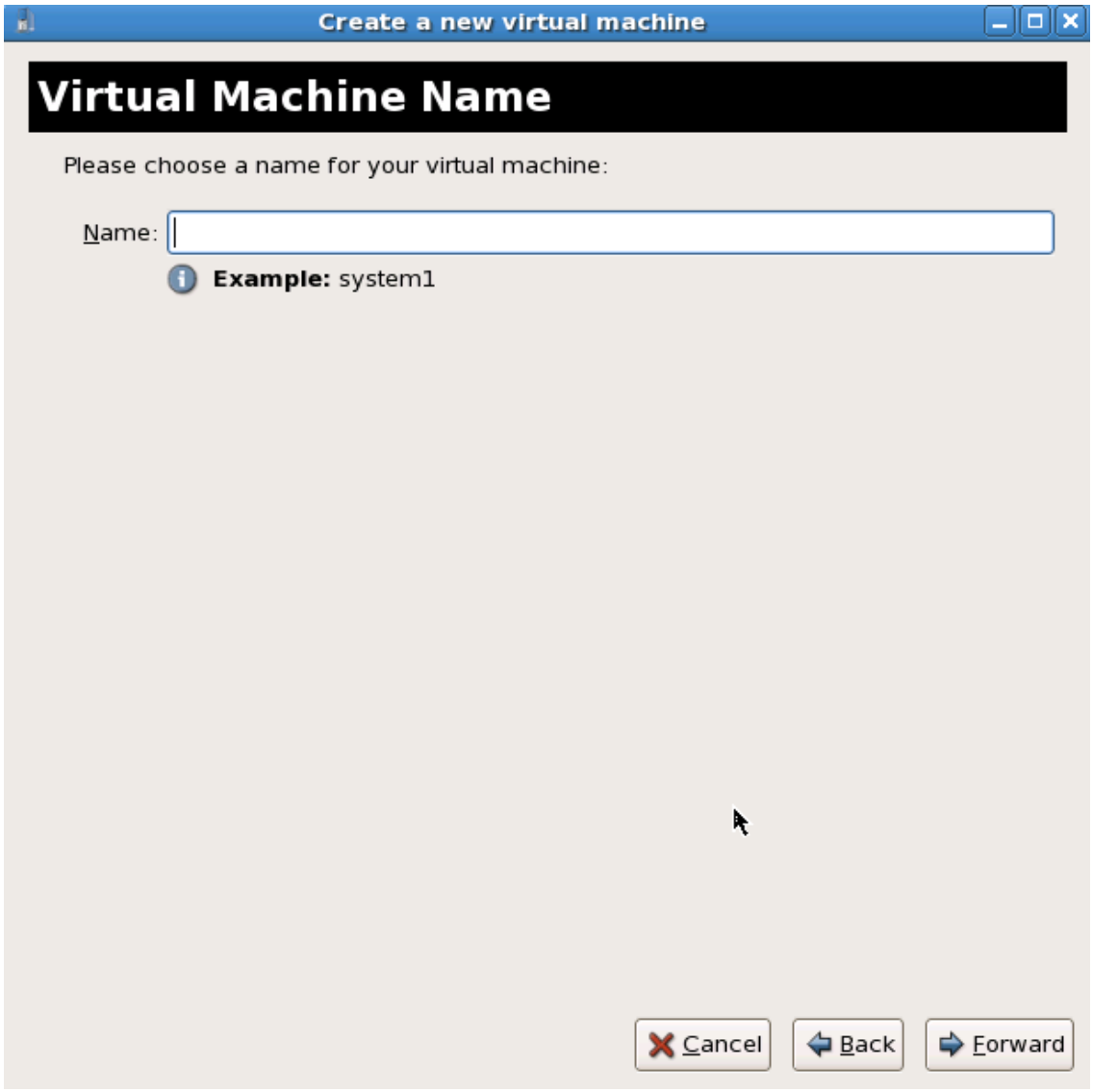
Kuvio 6: Uuden virtuaalikoneen luominen

Uusia virtualisia koneita voidaan luoda ja hallita Virtual machine manager-sovelluksella. Sovelluksen avulla voi myös seurata virtuaalikoneiden resurssikulutusta. Uusi virtuaalinen kone luodaan ”file”-valikosta valitsemalla kohta “New virtual machine”.



Kuvio 7: Alkuopastus

Tässä kohdassa kerrotaan yleisesti virtuaalikoneen asentamisesta ja sen vaiheista.

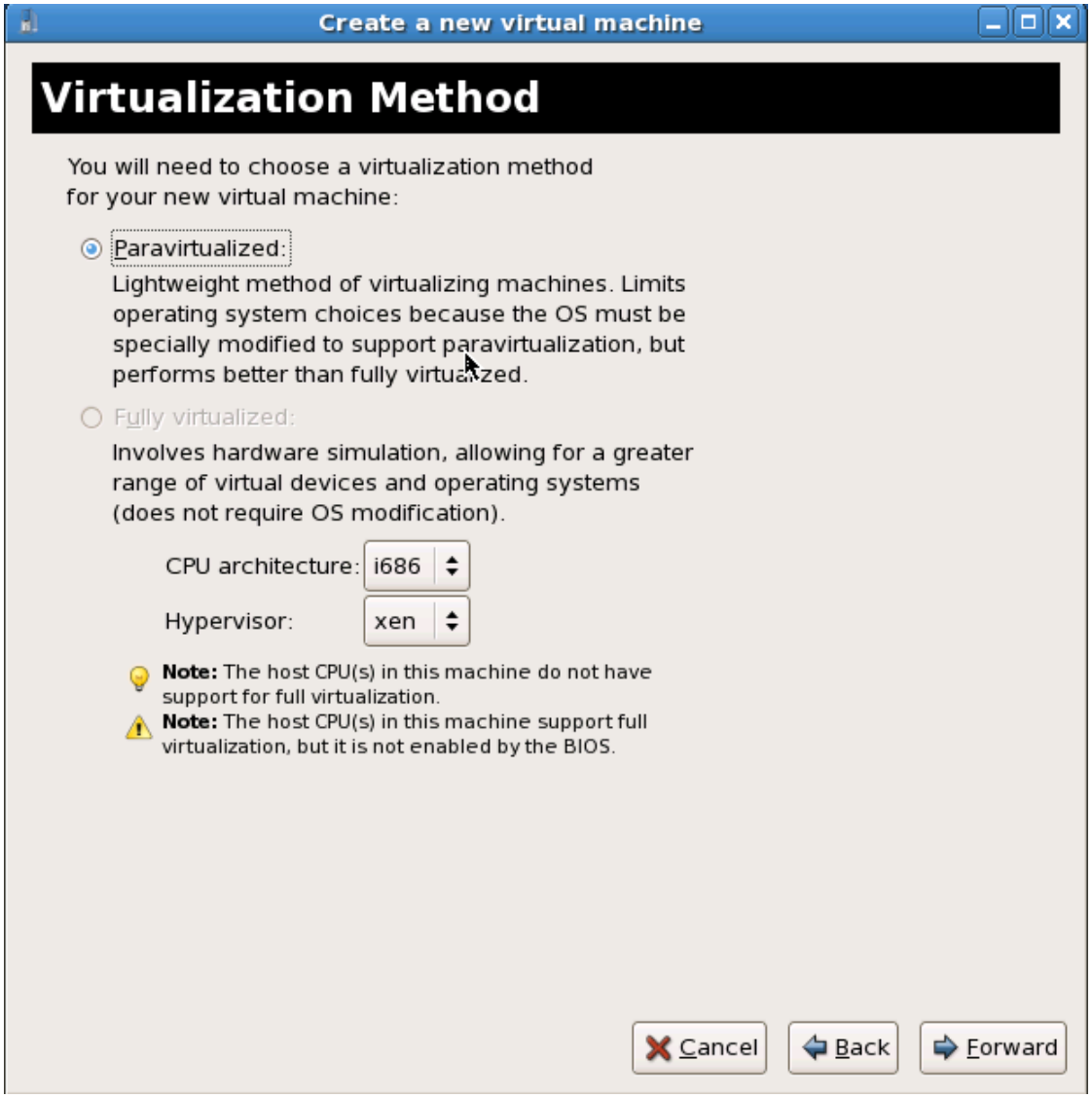


Kuvio 8: Virtuaalikoneen nimeäminen

Nimetään virtuaalikone (esimerkiksi lamp0). Nimeä voi muuttaa vielä asennuksen jälkeen terminaalissa konfiguraatiotiedostosta, joka sijaitsee polussa ”/etc/xen”. Tiedoston nimi on sama kun koneelle annettu nimi. Tiedostoa voi muokata esimerkiksi nano-tekstieditorilla. Esimerkki:

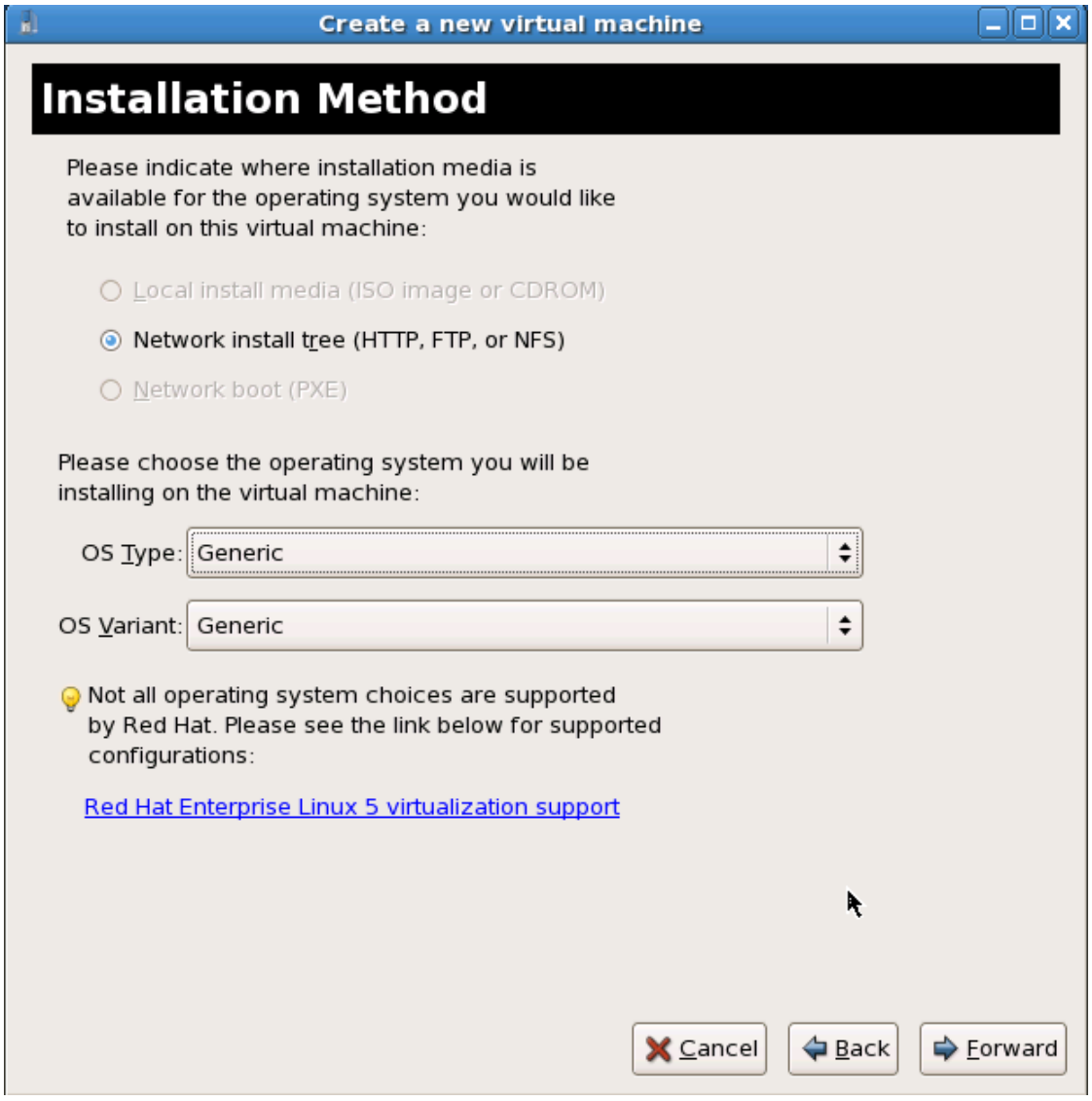
```
[root@localhost]#  nano lamp01
```

Kyseinen komento avaa konfiguraatiotiedoston lamp01. Konfiguraatiotiedosto sisältää mm. koneen nimen, id:n, enimmäis-muistimäärän, virtuaalilevykuvan polun ja muita tietoja.



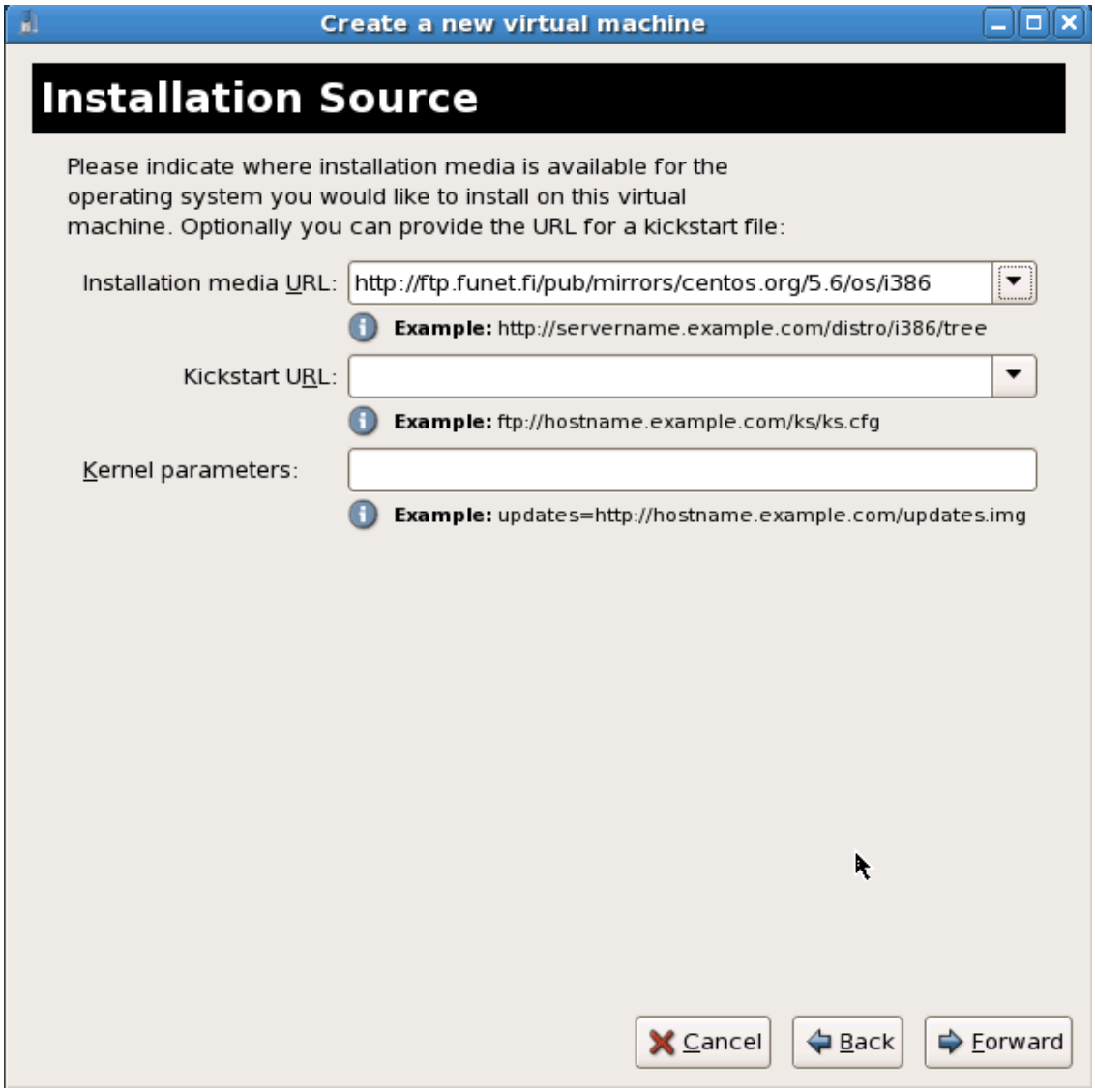
Kuvio 9: Asennusmetodi

Valitaan virtualisointi tapa, jota käytetään asennuksessa. Tässä kohdassa on valittu paravirtualisointi. Vaihtoehtoina on myös taysvirtualisointi, jolla voi simuloida fyysistä tietokonetta, mikä mahdollistaa esimerkiksi muiden käyttöjärjestelmien kuten Ms Windowsin asennuksen. Täysvirtualisointi vie suhteellisen paljon koneen resursseja, joten tässä projektissa se on otettu pois käytöstä bios-asetuksissa.



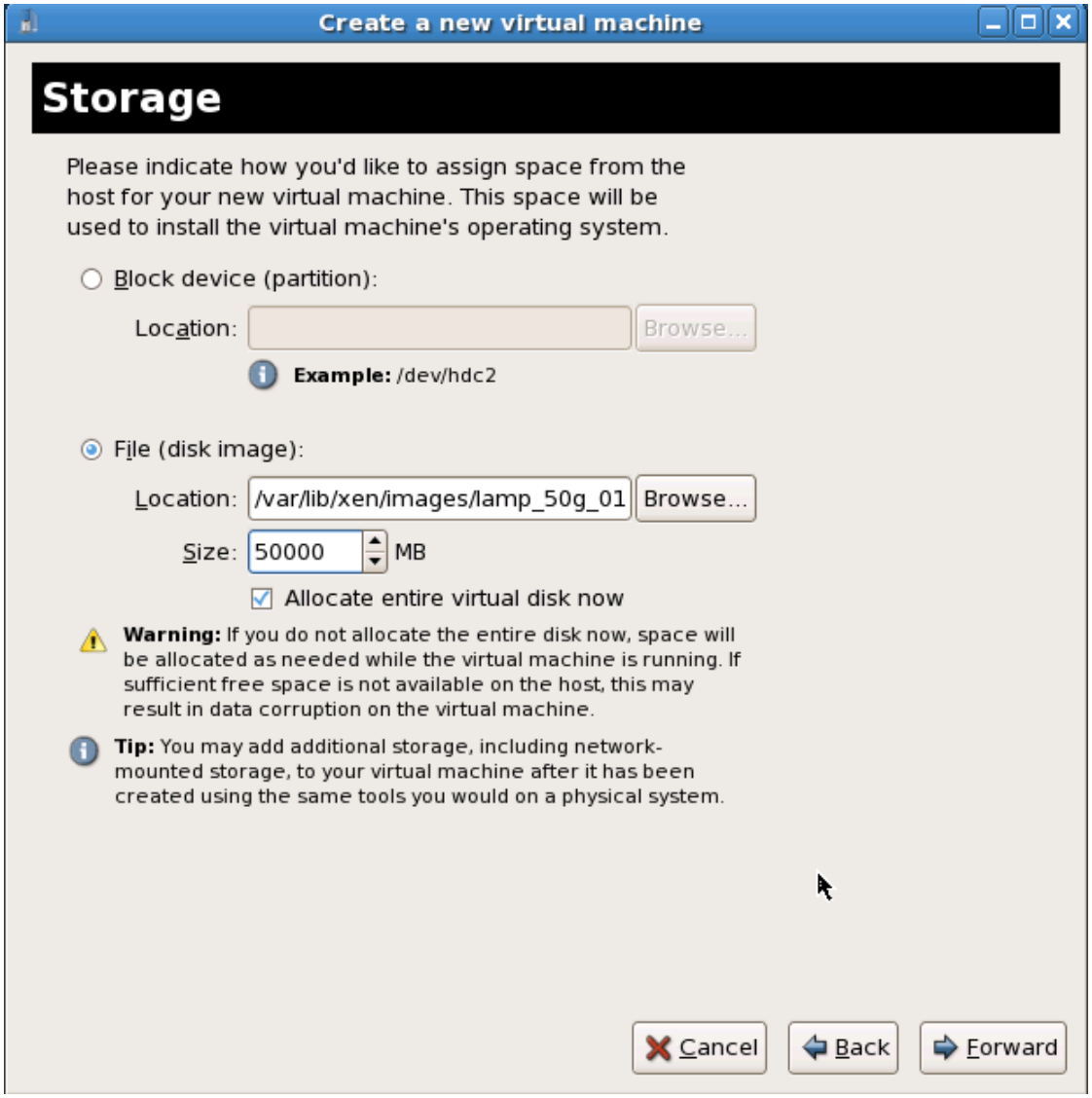
Kuvio 10: Asennusmetodin määritykset

Tässä kohdassa valitaan asennusmedian sijainti . Vaihtoehtona on levyasemalta asentaminen, http-, ftp- tai nfs-yhteydellä asentaminen. Muita vaihtoehtoja, joita ei käytetä tässä projektissa ovat asentaminen iso-levykuvasta tai optiselta asemalta sekä lähiverkon kautta käyttämällä PXE-protokollaa. Tässä valintaikkunassa valitaan myös käyttöjärjestelmä tyyppi ja variaatio, jotka voi jättää geneerisiksi, koska järjestelmä tunnistaa asennettavan käyttöjärjestelmän.



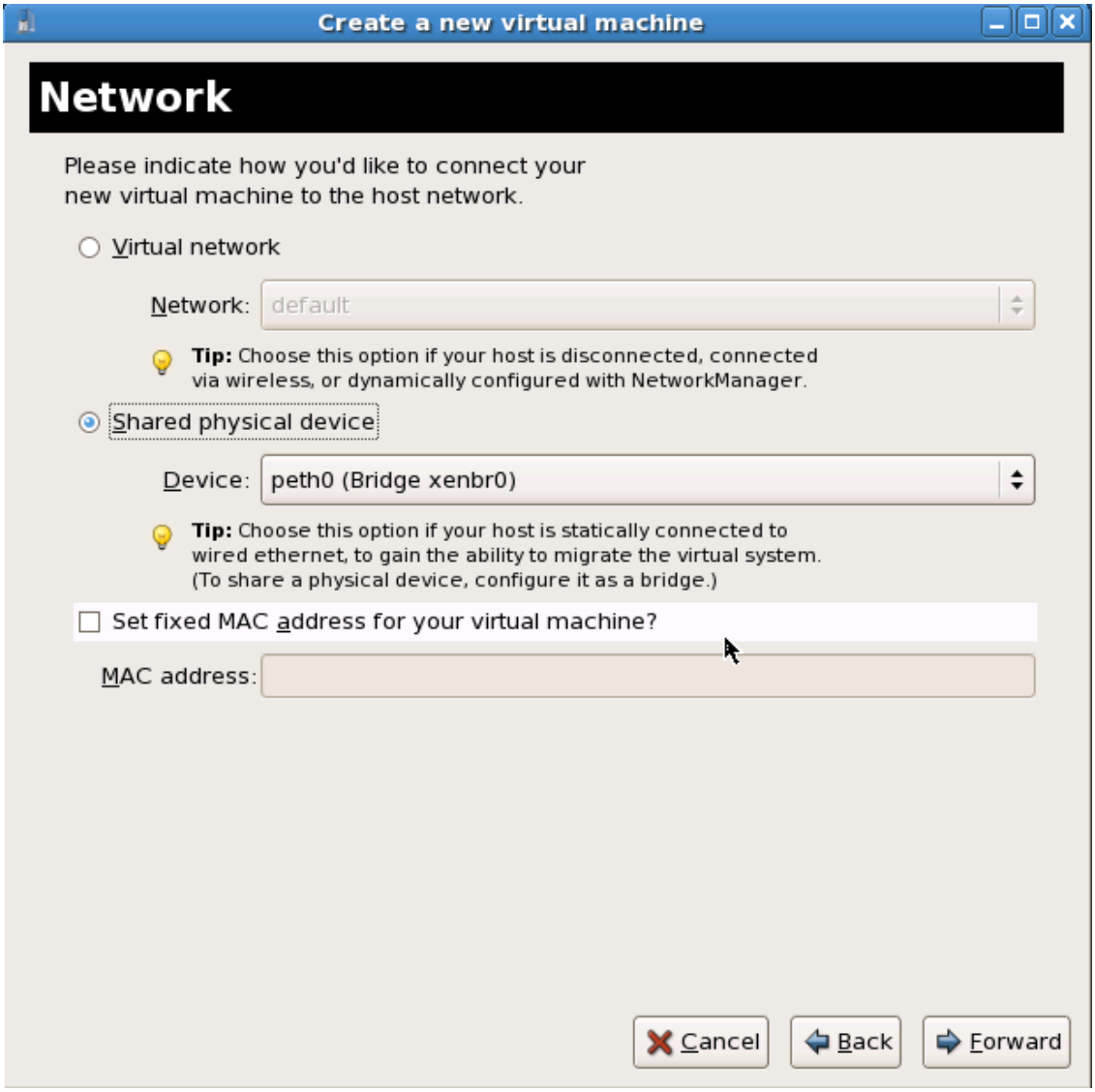
Kuvio 11: URL-osoitteen määrittäminen

Määritetään URL-osoite, josta asennus suoritetaan. Vaihtoehtoina on http-, ftp- ja .img-osoite kentät, joista asennus voidaan suorittaa. Listan asennus median url-osoitteista löytyvät centos.org sivulta valikosta downloads ja mirrors. Tässä esimerkissä käytämme asennus median url-osoitetta, joka sijaitsee suomalaisella funet.fi palvelimella. Asentaessa verkon kautta kannattaa aina valita maantieteellisesti lähin palvelin, jotta asennus sujuisi mahdollisimman nopeasi.



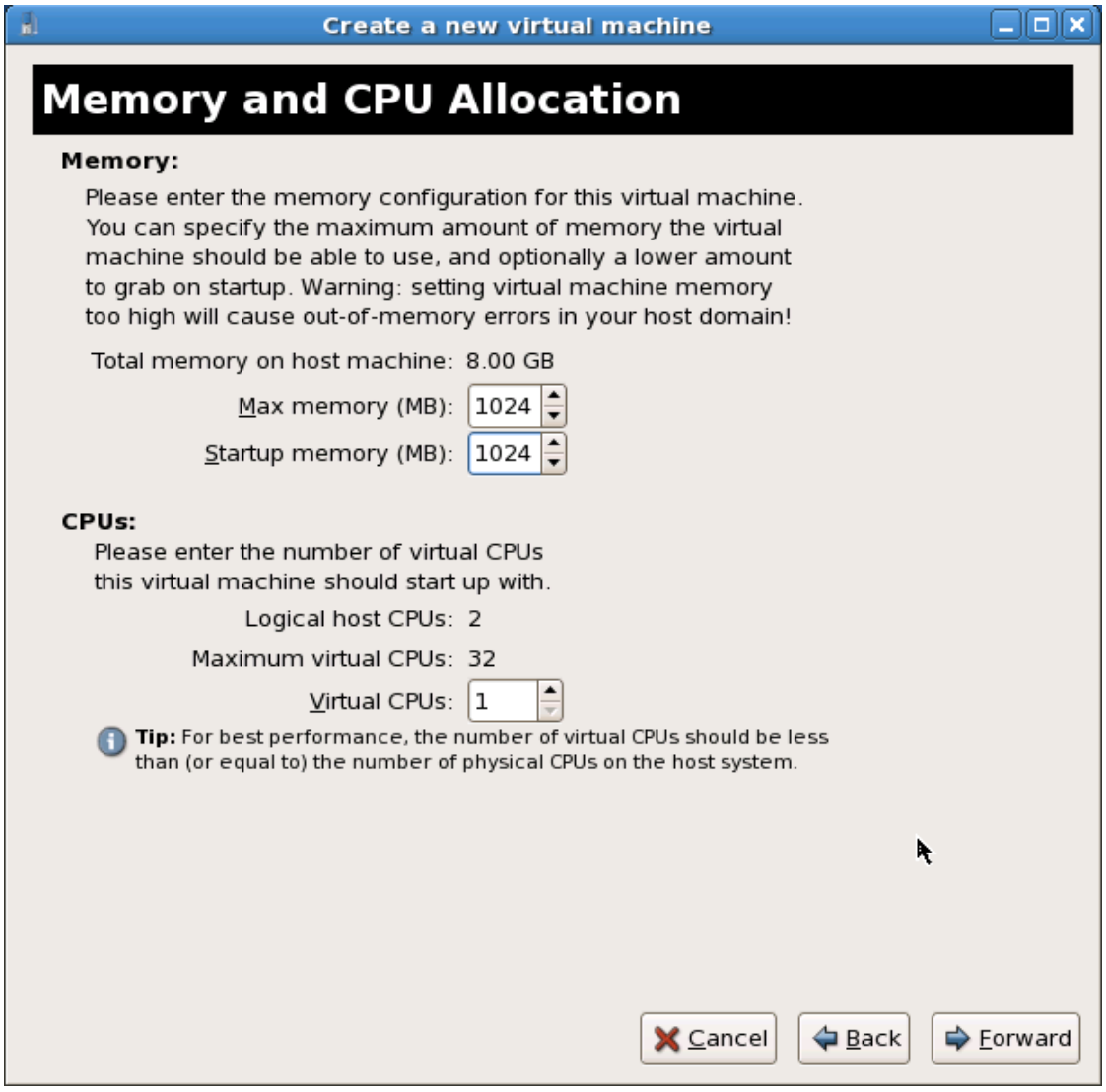
Kuvio 12: Levytilan valitseminen

Tässä kuviossa valitaan levytila. Vaihtoehtoina on uuden virtuaalisen palvelimen luominen fyysiselle lohkolaitteelle tai levykuvalle. Valintaikkunassa valitaan myös levytilan koko. Tässä kohdassa on valittu palvelimen luominen levykuvalla ja levytilaksi on määritelty 50 gigatavua. Levytilan koko valitaan tarpeen mukaan. Kannattaa ottaa huomioon, että itse käyttöjärjestelmä sekä LAMP-ohjelmistot vievät noin kaksi gigatavua tilaa. Tämän perusteella voidaan arvioida kuinka paljon tilaa tarvitaan. Tässä esimerkissä asennetaan isohkoa 50 gigatavun levytilalla varustettua virtuaalikonetta, jonka tilakapasiteetti riittää kymmenille käyttäjille ja niiden tiedostoille.



Kuvio 13: Verkon asetukset

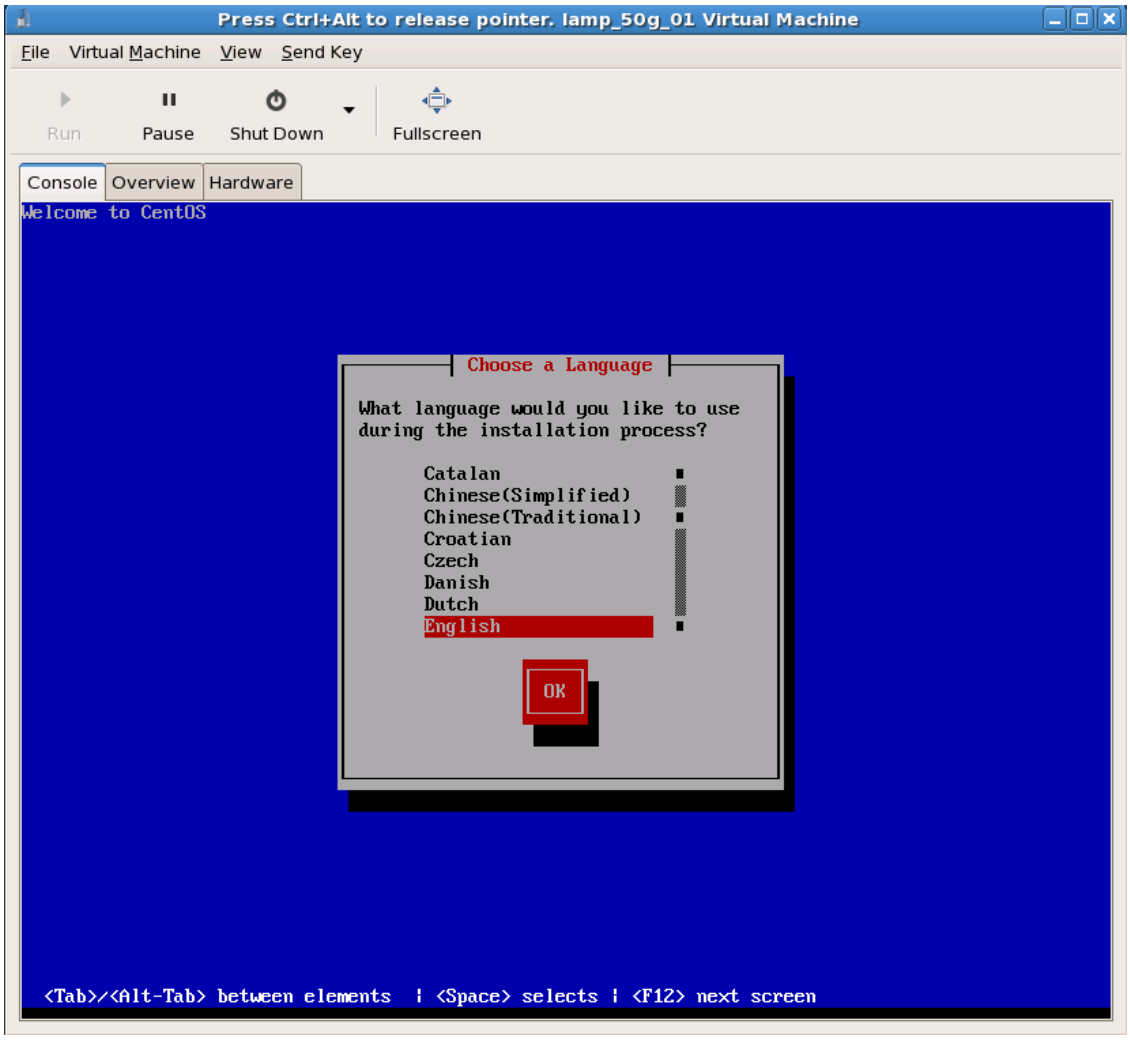
Valintaikkunassa määritetään miten virtualikone ottaa yhteyden isäntäverkkoon. Vaihtoehtoina on jaettu fyysinen verkkokortti tai virtuaaliverkko. Tässä projektissa olemme käyttäneet yleensä jaettua verkkokorttia sen joustavuuden takia. Virtuaalikone, jossa on jaettu verkkokortti on helpompi siirtää erimerkiksi toiseen xen isäntäkoneeseen tai ottaa talteen myöhempää käyttöä varten.



Kuvio 14: Muistinmäärä ja virtualisen suorittimen asetukset

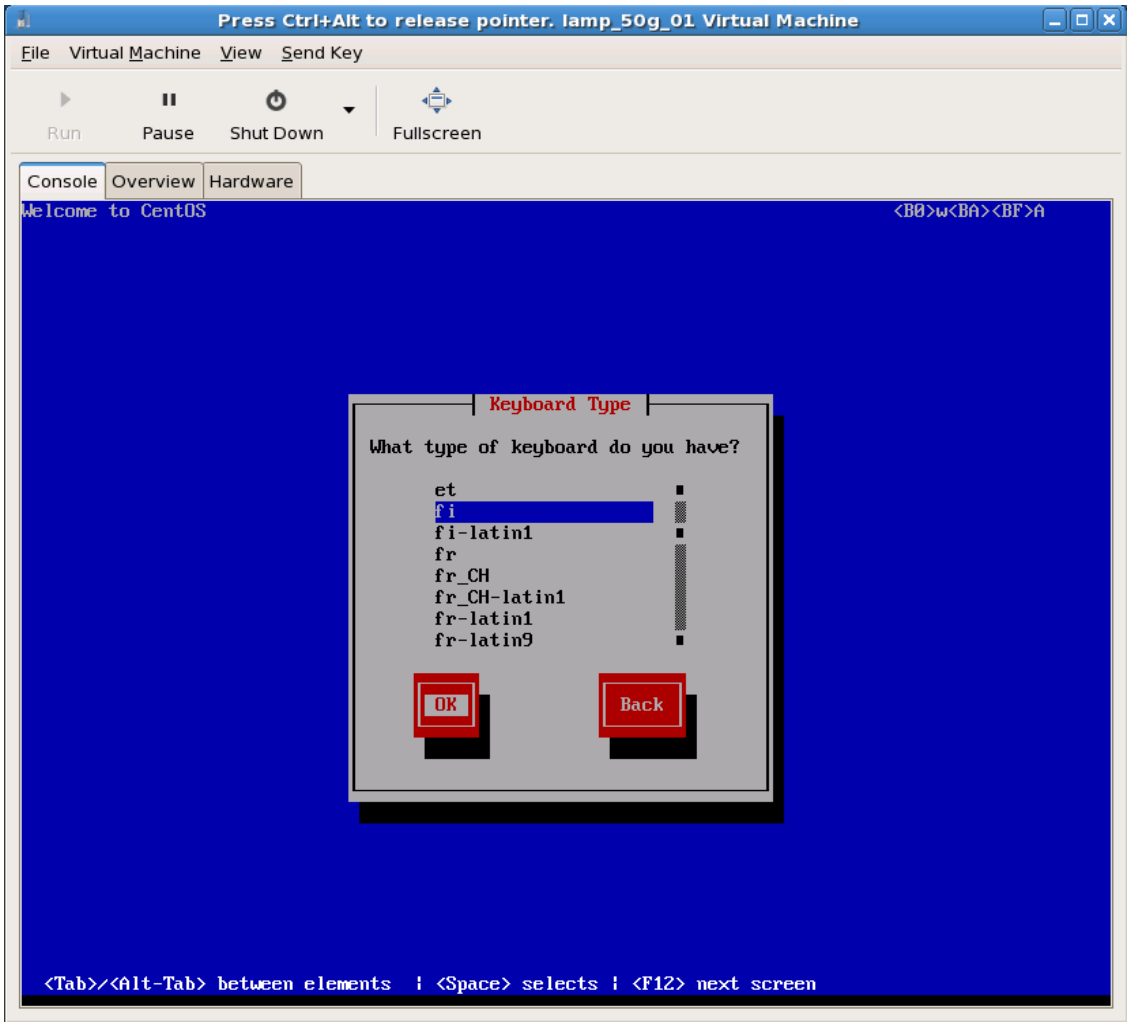
Valintaikkunassa valitaan virtuaalikoneelle muistin määrä. Tässä kohdassa tulee muistaa, että virtuaalikoneet käyttävät fyysisen koneen muistia jaettuna resurssina. Kyseisessä koneessa on ainoastaan 8 gigatavua muistia. Eli kyseisellä muistin määrällä voi luoda seitsemän virtuaali konetta, jotka käyttävät muistia 1024 megatavua. Pienissä palvelin projekteissa voi käyttää myös vähemmän muistia, esimerkiksi 512 megatavua on riittävä määrä lamp-palvelimen konfiguraation ajamiseen.

Tässä konfiguraatio dialogissa voi myös valita virtualisen prosessorien määrän. Fyysisessä koneessa on tällä hetkellä Intel xeon prosessori, jossa on kaksi ydintä. Parhaimman teho-hyöty suhteen saavuttamiseksi ei kannata valita enempää suorittimia kuin mitä on fyysisella koneella. Suurimmassa osassa tapauksista yksi virtuaalisuoritin per virtuaalikone on riittävä.



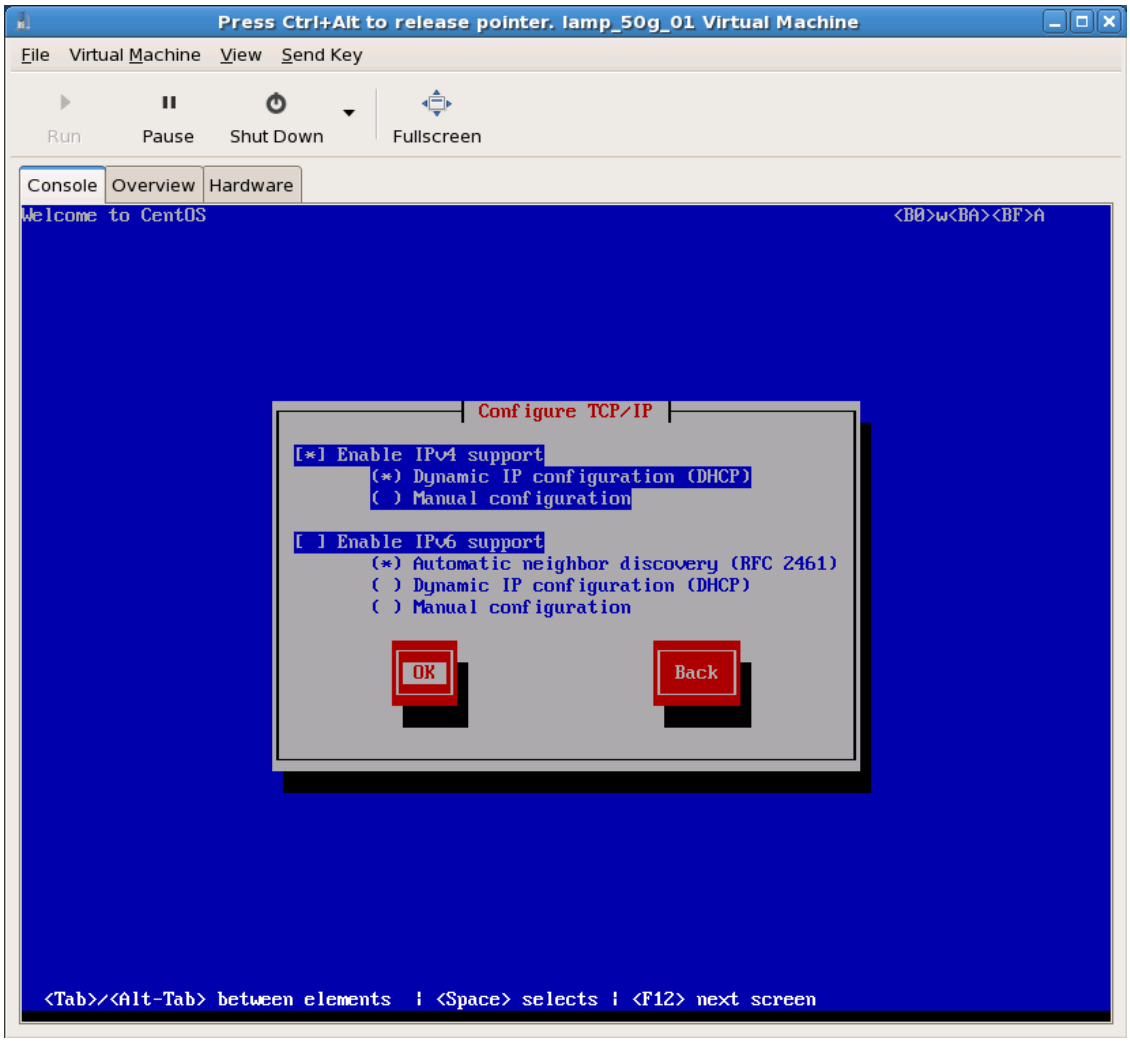
Kuvio 15: Kieliasetukset

Valitaan kieli asennusprosessin ajaksi ja edetään seuraavaan vaiheeseen. Kielivalinta ei ole lopullinen.



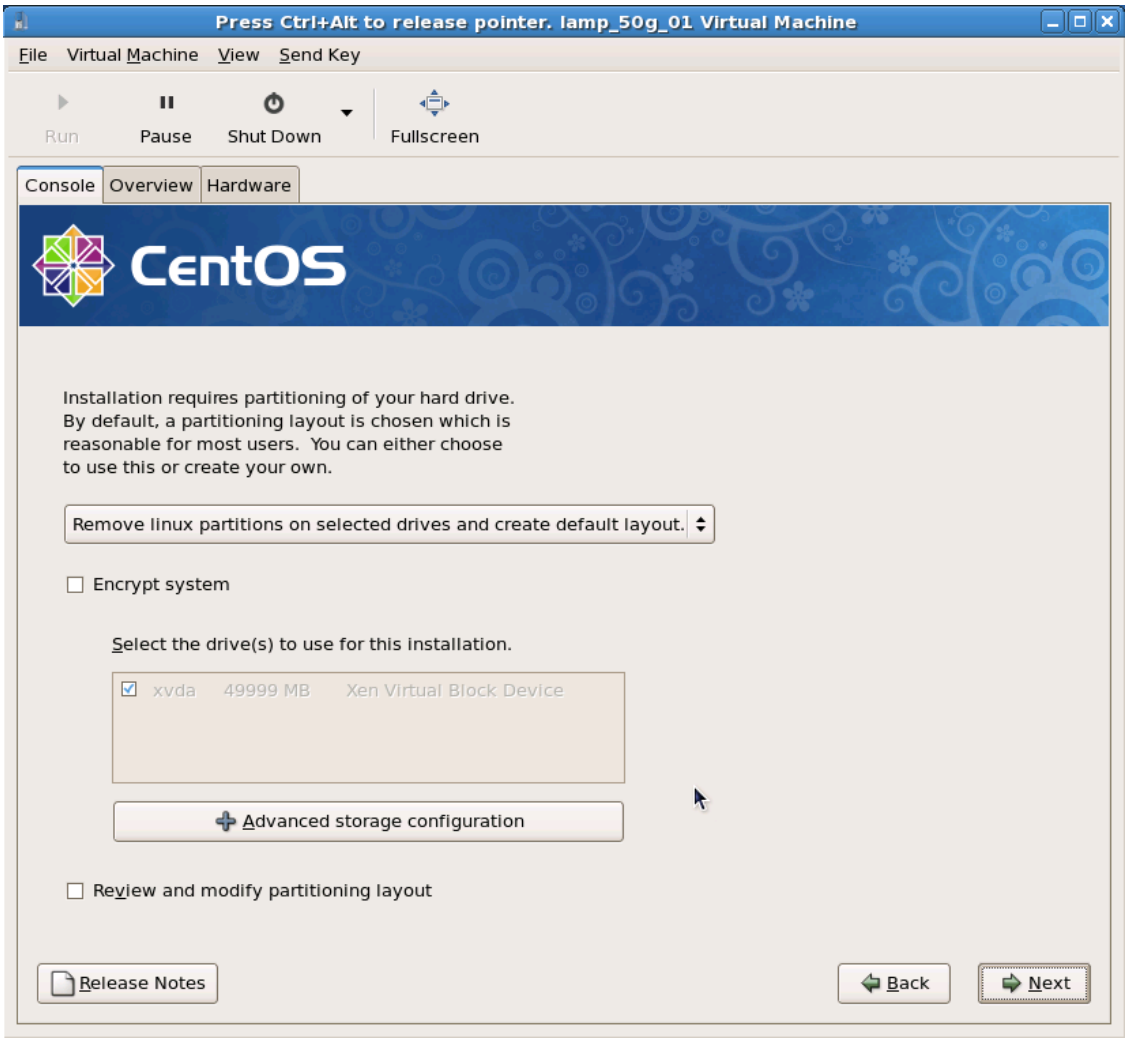
Kuvio 16: Näppäimistön valitseminen

Tässä valintaikkunassa valitaan mitä näppäimistöä halutaan käyttää asennuksen aikana. Oikean näppäimistön valitseminen on tärkeätä, jotta asennus sujuisi helpommin.



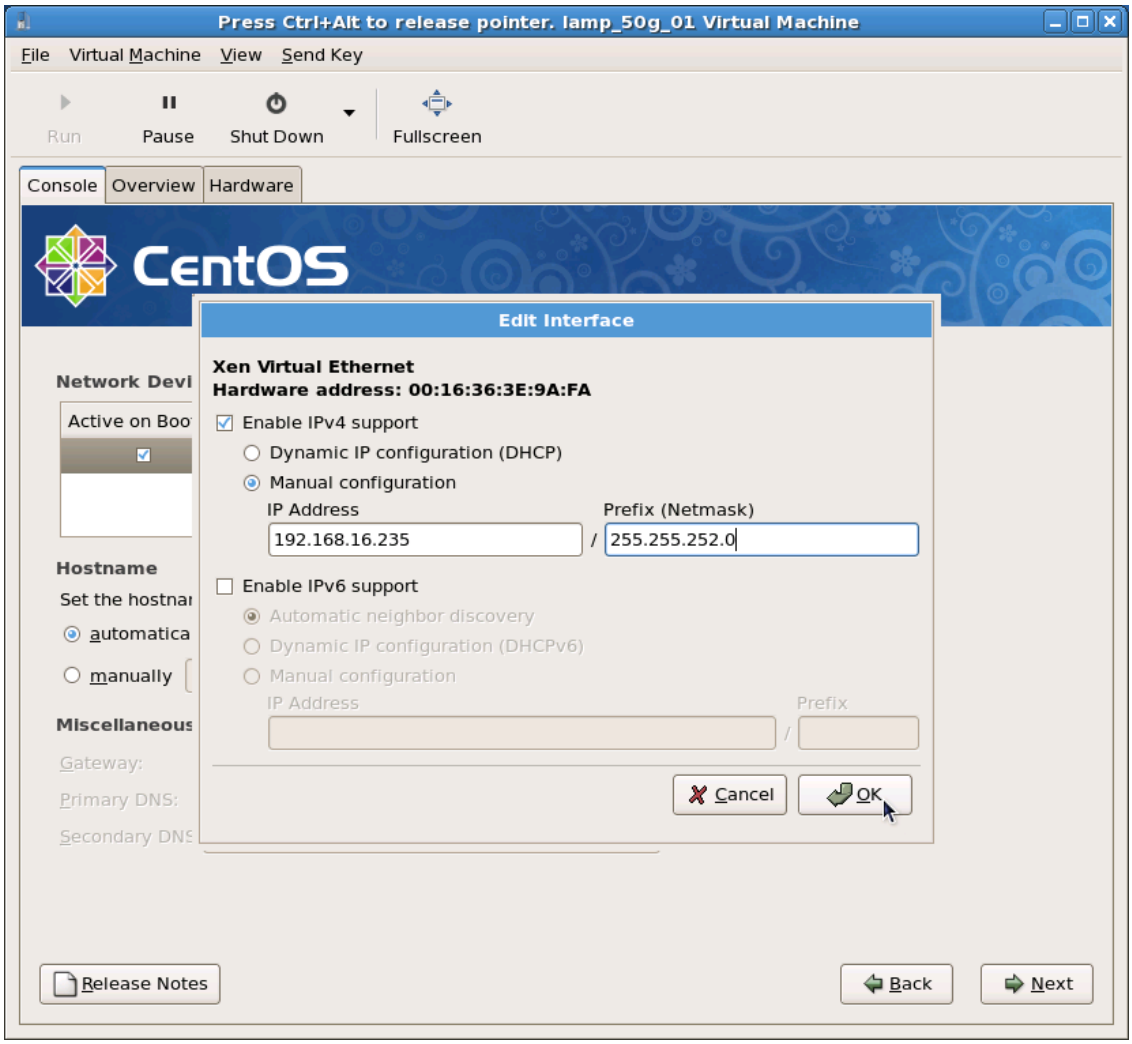
Kuvio 17: Ipv4- ja Ipv6-asetusten määrittäminen

Valintaikkunassa määritetään IPv4- ja IPv6-asetuksia. IPv4 kohdassa on mahdollista määrittää automaattinen DHCP-konfiguraatio tai manuaalisesti määriteltävä ip-osoite. IPv6 valikossa vaihtoehdot ovat samat. Virtuaali palvelimen alkuasennuksen aikana ipv4-verkkoasetukset haetaan automaattisesti dhcp-protokollalla ja myöhemmin vaihdetaan manuaalisiksi. Ipv6-verkonasetukset otetaan pois käytöstä (*-merkki pois kohdasta ”enable IPv6 support”), koska ipv6-protokolla ei ole tuettuna tl-labolatorion verkossa.



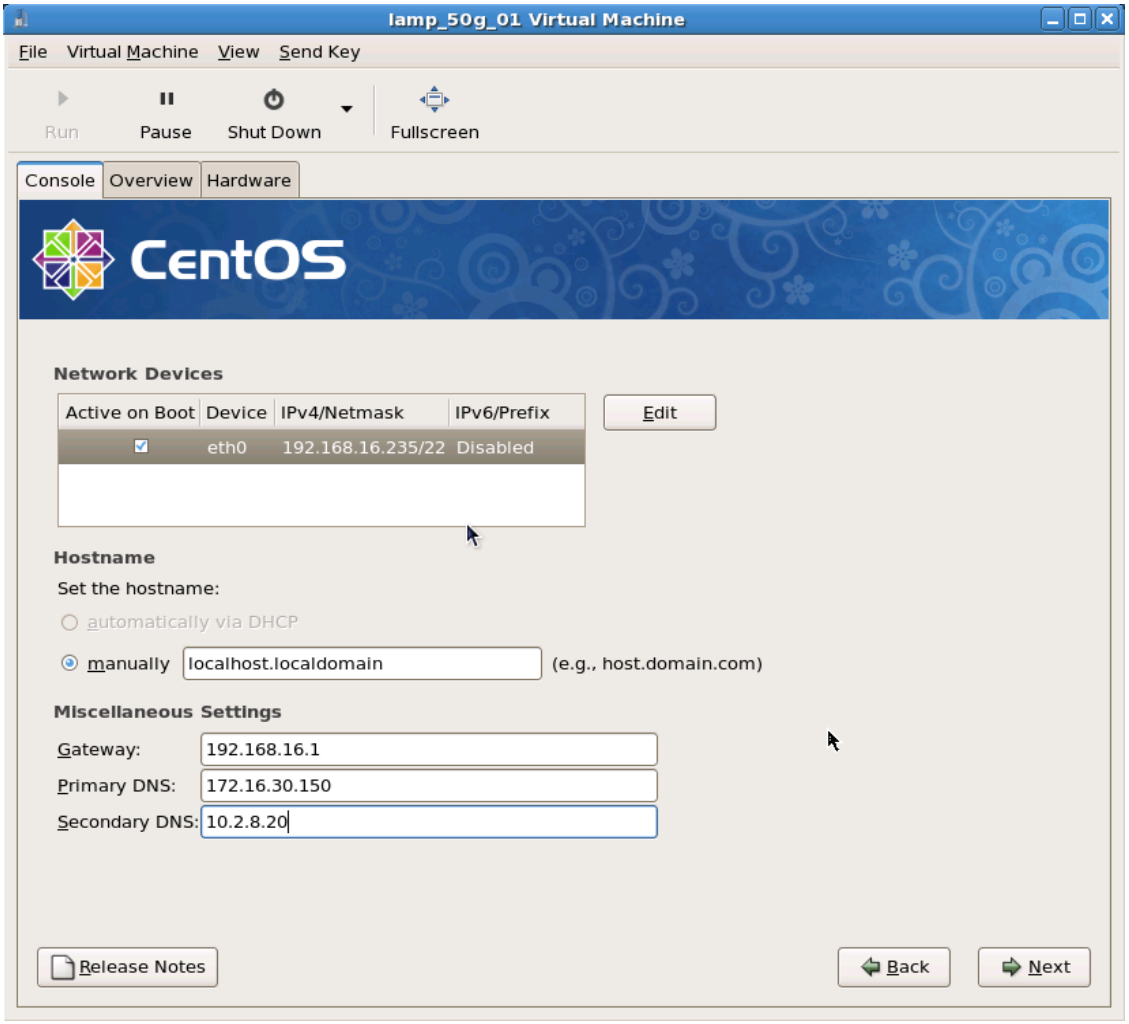
Kuvio 18: Kovalevy asetusten määrittäminen

Valintaikkunassa määritetään kovalevy, jota käytetään käyttöjärjestelmän asennukseen. Tässä tapauksessa käytetään xen virtuaalista levykuvaa (xvda). Pudotusvalikosta valitaan kohta jossa poistetaan kaikki levyjaot (partitions) valituissa levyissä ja käytetään oletus levyjako skenaariota asennukseen. Tämän jäkeen asennusdialogi kysyy varmistusta ennen kuin kovalevy tyhjennetään ja jaetaan. Järjestelmässä voi myös käyttää monimutkaisempaa useamman partition skenariota, mutta olemme todenneet että yksinkertaisempi levynjako skenarion on helpompi ylläpitää.



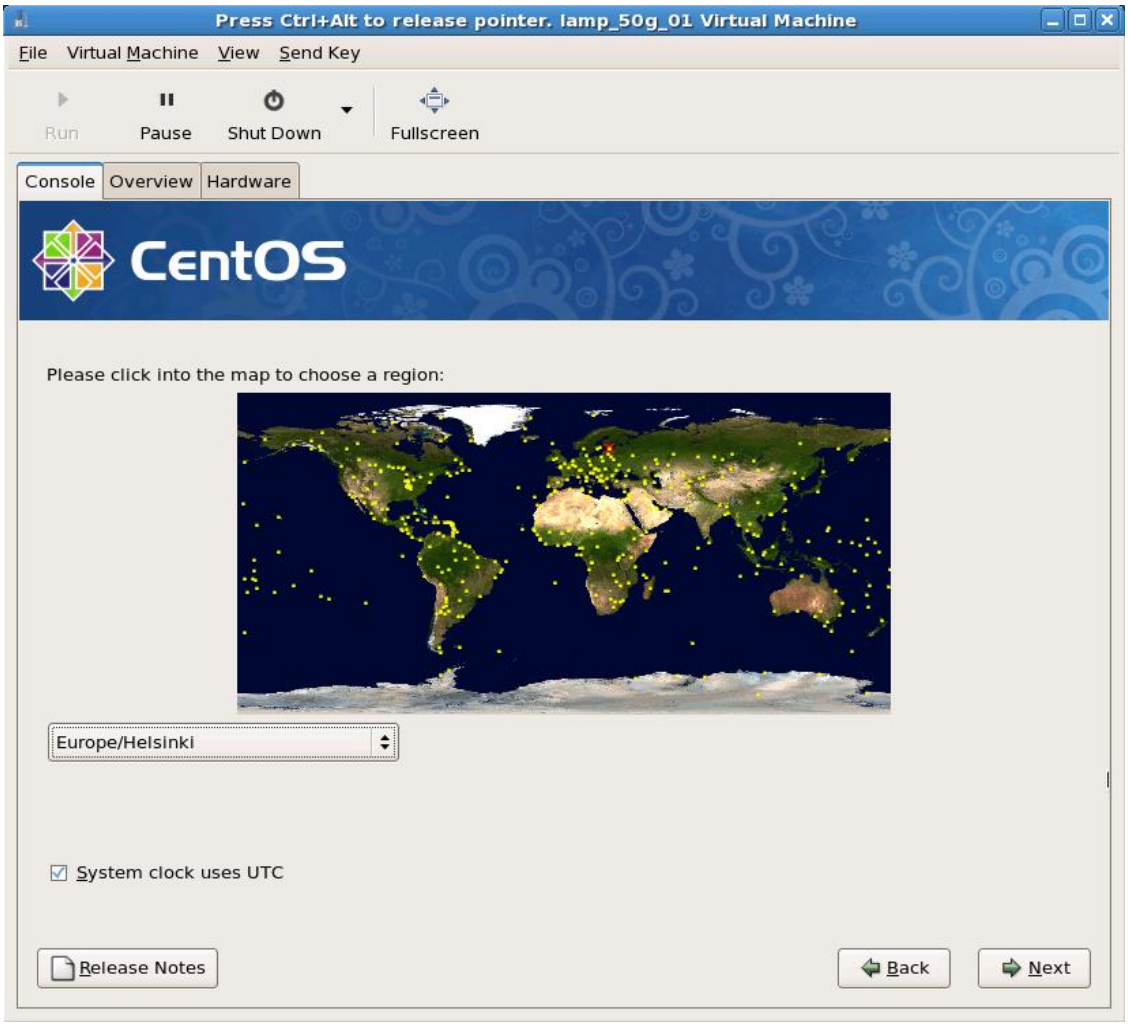
Kuvio 19: Ipv4-osoitteen manuaalinen asentaminen

Valintaikkunassa määritetään manuaalisesti IPv4 ip-osoite ja aliverkonpeite. Vaihtoehtona on myös dynaamisen ip-osoitteen konfigurointi. Tällä hetkellä virtuaaliselle palvelin ympäristölle on jaettu ip osoiteavaruus 192.168.16.225 – 254. Ennen ip-osoitteen valintaa on suositeltavaa tarkistaa, mitkä osoitteet ovat jo käytössä ASA-palomuurin asetuksista tai kysyä tl-labolatorion henkilökunnalta.



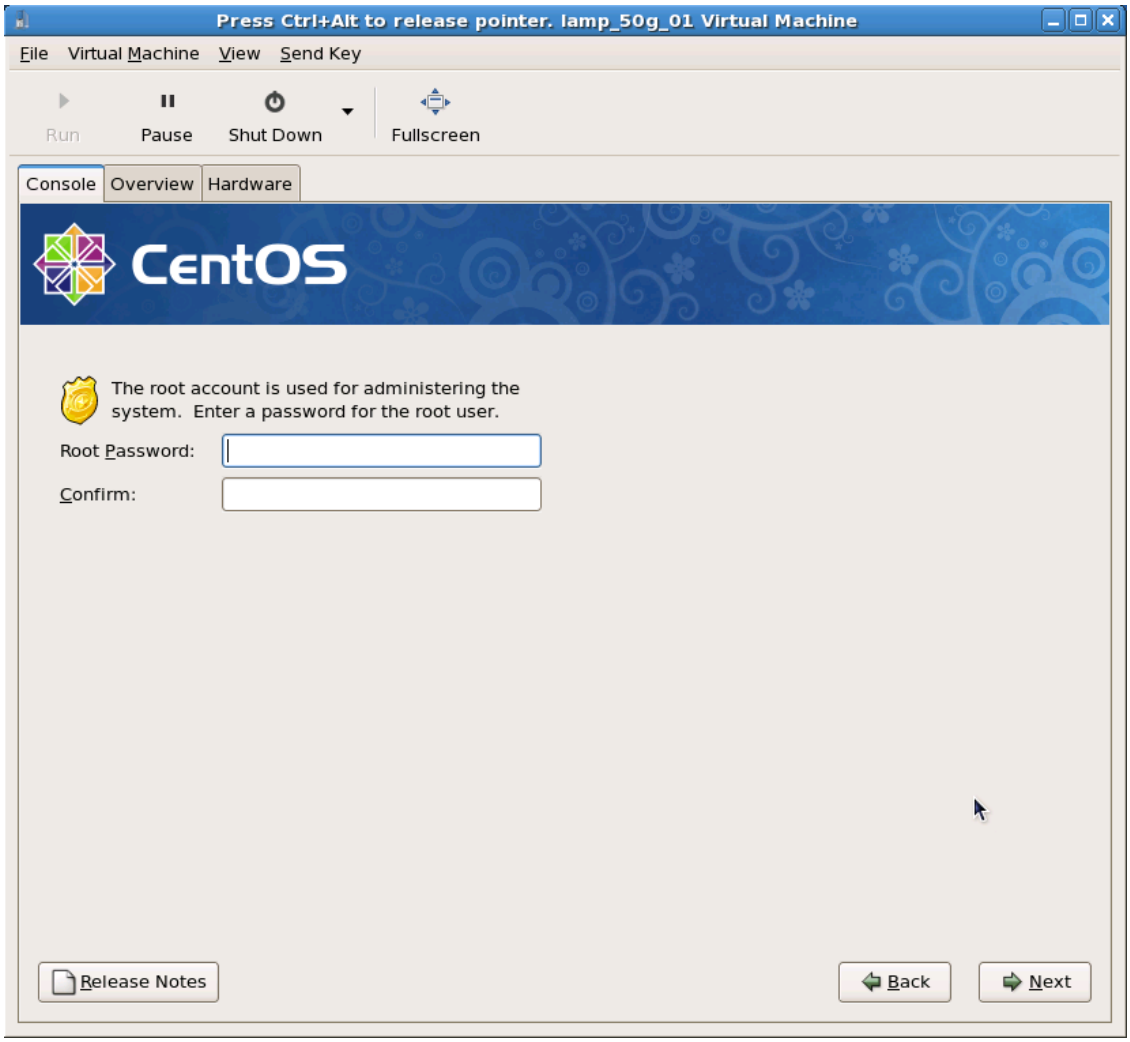
Kuvio 20: Osoitteen nimeäminen ja nimipalvelin osoitteiden määrittäminen

Tässä valintaikkunassa määritetäään ”hostname” ja Dns:n eli nimipalvelimien ip-osoitteet. Nimipalvelinten vaihtaminen on mahdollista myös jälkikäteen. Dns-osoitteet ovat Kuvion 20 mukaiset palvelinosoitteet.



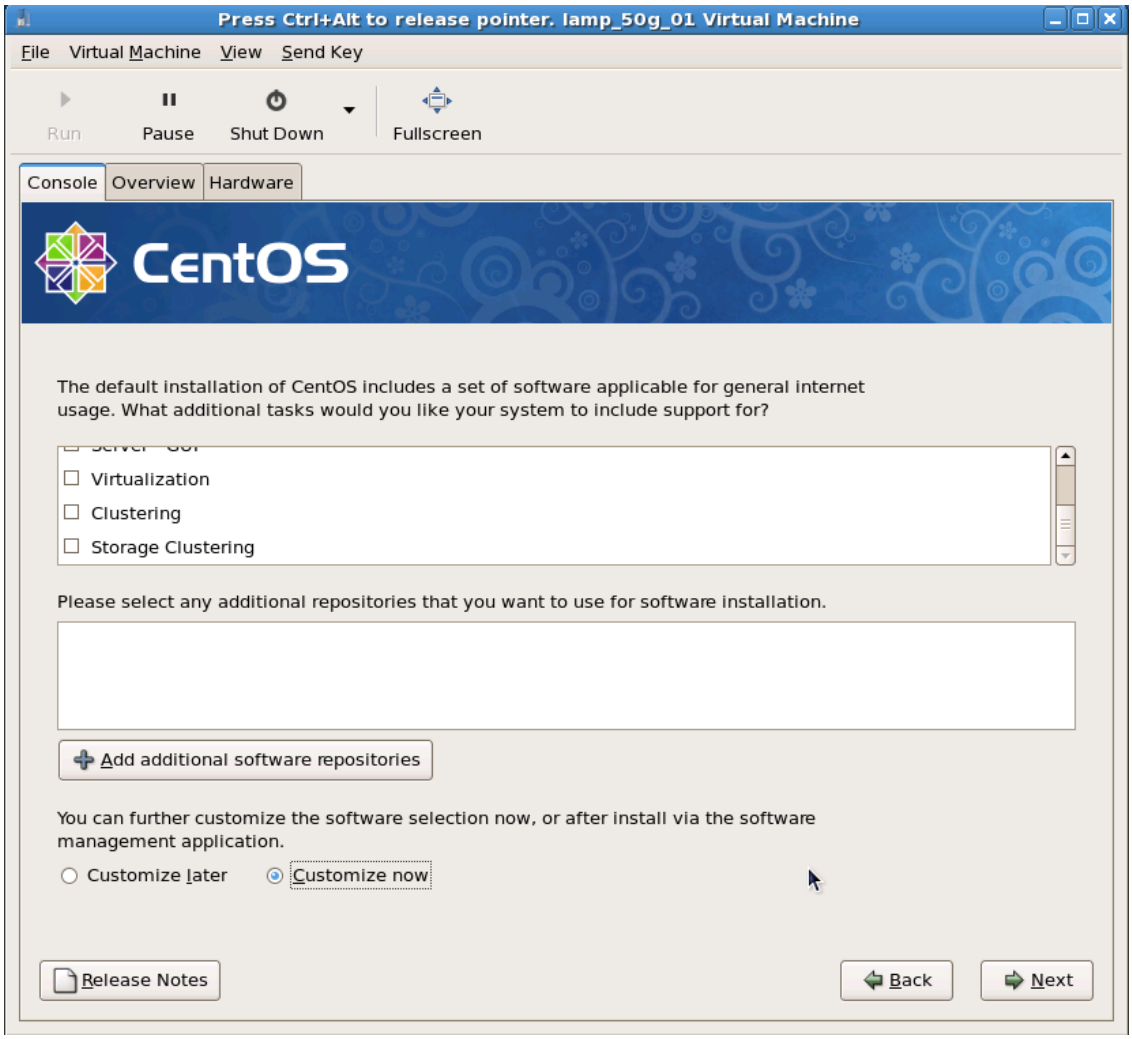
Kuvio 21: Aikavyöhyke

Tässä asennuksen vaiheessa valitaan aikavyöhykkeeksi Europe/Helsinki ja asetetaan rasti kohtaan System clock uses UTC.



Kuvio 22: Pääkäyttäjän tunnukset

Valintaikkunassa luodaan pääkäyttäjän tunnukset. Unix-pohjaisissa järjestelmissä pääkäyttäjän nimi on aina ”root”. Käyttäjätunnus kannattaa määrittää huolella ja sen tulee sisältää vähintään kahdeksan merkkiä, joissa on isoja ja pieniä kirjaimia sekä numeroita.

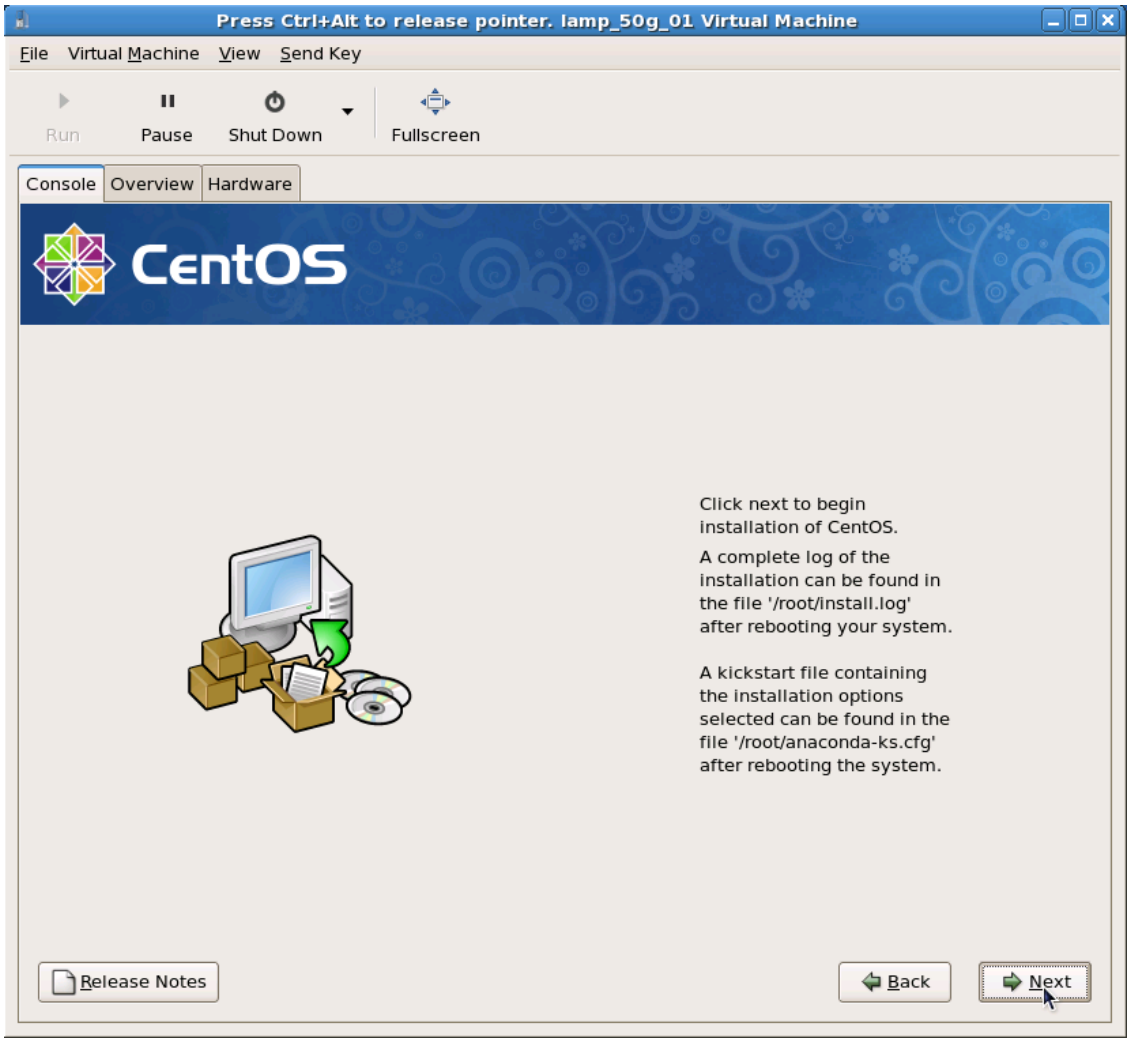


Kuvio 23: Pakettikokonaisuudet

Tässä ikkunassa valitaan pakettikokonaisuudet, joita käytetään järjestelmän asentamiseen. Valintaikkunassa on rastitettu pois kaikki pakettikokonaisuudet. Tämän jälkeen alemmassa kohdassa valitaan ”Customize now” ja siirrytään seuraavaan asennusvaiheeseen kohdasta ”Next”, jossa määritetään tarkemmat yksittäiset ohjelmistopaketit. Kyseinen toimenpide tehdään, koska järjestelmä asentaa oletuksena hyvin paljon ylimääräisiä paketteja, joita ei tarvitse palvelimen ajamiseen. Järjestelmä asentaa esimerkiksi graafisen käyttöliittymän, pelejä, nettiselaimia ja muita ylimääräisiä paketteja, jotka voivat aiheuttaa kuormitusta ja tietoturvariskejä.

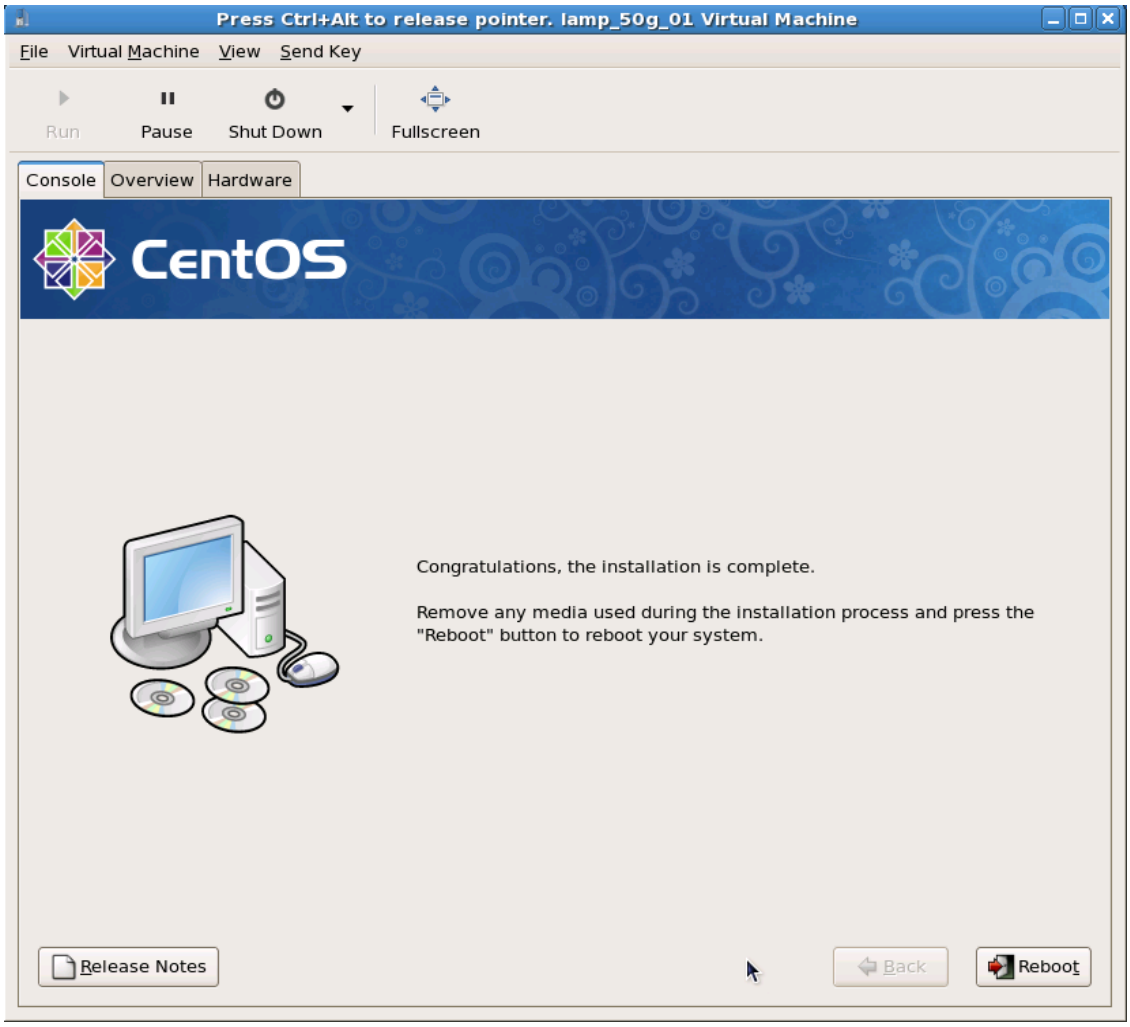
Kuvio 24: Ohjelmistopaketit

Ikkunassa valitaan yksittäiset ohjelmistopaketit, joita tarvitaan esimerkiksi Lamp-palvelimen ajoon ja ylläpitoon. Tässä vaiheessa asentajan kannattaa miettiä pakettivalintoja, koska tarkoituksena on pitää järjestelmä mahdollisimman yksinkertaisena ja tarkoituksen mukaisena.



Kuvio 25: Asennusloki

Kohdassa ”Next” siirrytään ajamaan asennus loppuun. Asennuslokitiedostoa voi katsoa jälkikäteen terminaalissa hakemistopolusta:”/root/install.log” .



Kuvio 26: Järjestelmän uudelleenkäynnistys

Asennuksen päätteeksi järjestelmä käynnistetään uudelleen, jolloin asetukset astuvat voimaan.