

## Raspberry Pi 2 -tietokoneen soveltuvuus palvelimeksi

Antero Virto





<b>Tekijä</b> Antero Virto	
<b>Koulutusohjelma</b> Tietojenkäsittelyn koulutusohjelma	
<b>Opinnäytetyön otsikko</b> Raspberry Pi 2 -tietokoneen soveltuvuus palvelimeksi	<b>Sivu- ja liitesivumäärä</b> 36 + 1
<b>Opinnäytetyön otsikko englanniksi</b> Raspberry Pi 2 as a server	
<p>Raspberry Pi on edullinen yhden piirilevyn tietokone, joka on noin luottokortin kokoinen. Sen uusin versio, Raspberry Pi 2 model B ("RPI2") maksaa noin 50€, ja sille on saatavilla käyttöjärjestelmäksi Debian-nimiseen Linux-jakeluun pohjautuva ilmainen Raspbian.</p> <p>Koska itse tietokone on edullinen ja ilmaiselle Raspbianille on saatavilla suuri osa palvelinkäytössäkin suositun Debianin sovellusvalikoimista, se on mielenkiintoinen mahdollisuus kevyeksi palvelinkoneeksi, esimerkiksi pienelle yritykselle tai kolmannelle sektorille. Tässä opinnäytetyössä tutkitaan RPI2:n soveltuvuutta juuri tuohon tarkoitukseen, kiinnittäen huomioita sen suorituskykyyn, sovellusvalikoimaan ja sovellusten toimintaan käytännössä.</p> <p>Työn teoriaosuudessa tarkastellaan RPI2:hta fyysisenä tietokoneena, huomioiden sen komponentit kuten prosessori, keskusmuisti ja tiedon tallennusmahdollisuus. Raspbian-käyttöjärjestelmään perehdytään ja tarkastellaan sen sovellusvalikoimaa. Lopuksi tarkastellaan tutkittaviksi valittuja palvelinsovellustyyppisiä ja sovelluksia (DHCP, HTTP ja nimipalvelu).</p> <p>Työn empiirisessä osuudessa kuvataan ja pystytetään testiympäristö ja suoritetaan valittujen palvelinsovellusten testaus. RPI2 liitetään muutaman muun tietokoneen kanssa lähiverkkoon, jotta valittuja sovelluksia (DHCP, nimipalvelu) voidaan testata käytännössä ja arvioida niiden käyttöön helpoutta ja toimintaa. Lisäksi sen suorituskykyä arvioidaan HTTP-raitustestauksella. Lopuksi arvioidaan saatuja tuloksia ja niiden kautta tutkimuskysymystä, eli RPI2:n soveltuvuutta palvelinkäyttöön.</p>	
<b>Asiasanat</b> Palvelin, RISC, ARM, Raspbian, Raspberry Pi, RPI2	

<b>Author</b> Antero Virto	
<b>Degree programme</b> Business Information Technology	
<b>Report/thesis title</b> Raspberry Pi 2 as a server	<b>Number of pages and appendix pages</b> 36 + 1
<p>The Raspberry Pi is an inexpensive single-board computer, roughly the size of a mobile phone. The newest configuration, Raspberry Pi 2 model B ("RPI2"), costs about 50€ in Finland, and its recommended Linux operating system, Raspbian, is free. Considering that Raspbian's parent distribution, Debian, is used as a server operating system, and that the hardware itself is low-cost, running a lightweight server setup on the RPI2 might be an interesting possibility for small businesses or the third sector. This thesis examines the RPI2's suitability for server use, focusing on what kind of server use could the RPI2 theoretically be fit for, considering its performance and other hardware limitations and the selection of server software.</p> <p>In the first part of the thesis, the RPI2 is examined as a physical computer, taking into account its amount of memory, computational power and data storage capability. Raspbian is also taken a closer look at: its relation to the more popular Debian operating system and what it means regarding the selection of software and the frequency of updates. Finally, various types of servers (DHCP, DNS and HTTP) are introduced as possible candidates for running on the RPI2, weighing the pros and cons of running each on such frugal hardware.</p> <p>The second, empirical part of the thesis describes setting up the RPI2 and a testing environment, and the testing itself. This essentially means connecting the RPI2 to a local area network with various other computers, installing and setting up the selected server programs and seeing how they work in a real world scenario. HTTP server benchmarking is used to get an estimate of the RPI2's performance, while the DHCP and DNS servers are set up to see if they install properly and do what they're supposed to. The thesis concludes with analysis of the results and thoughts on whether the RPI2 is indeed suitable for server use, and thoughts on the thesis process and the learning experience.</p>	
<b>Keywords</b> server, RISC, ARM, Raspbian, Raspberry Pi, RPI2	

## Sisällys

1 Johdanto.....	1
2 Työn tavoitteet ja rajaus.....	3
3 Tutkittava tietokone.....	4
3.1 Tietokoneen yleiskatsaus.....	4
3.2 Suoritin.....	5
3.3 Tiedon tallennus.....	5
4 Sovellukset.....	6
4.1 Raspbian-käyttöjärjestelmä.....	6
4.2 Palvelinsovellukset.....	6
4.2.1 DHCP.....	7
4.2.2 Nimipalvelin.....	7
4.2.3 HTTP-palvelin.....	8
5 Sovellusten asennus ja ympäristö.....	10
5.1 Testausympäristön kuvaus.....	10
5.2 RPI2:n fyysinen kokoonpano.....	11
5.3 Muistikortille asennus ja varmuuskopiointi.....	11
5.3.1 Levykuvan tiputus muistikortille.....	11
5.3.2 Asennetun järjestelmän varmuuskopiointi levykuvaksi.....	12
5.4 Käyttöjärjestelmän käyttöönotto ja määrittelyt.....	12
5.4.1 Etähallinta SSH:lla.....	12
5.4.2 Staattinen IP-osoite.....	13
5.4.3 Muut asetukset ja raspi-config.....	13
5.5 Palvelinsovellukset.....	14
5.5.1 dnsmasq.....	14
5.5.2 BIND9.....	15
5.5.3 ISC DHCP.....	17
5.5.4 nginx.....	19
5.5.5 Apache2.....	20
5.5.6 Varnish ja Apache2.....	20
5.5.7 Wordpress.....	21
6 HTTP-palvelinten suorituskyvyn testaus.....	23
6.1 ab: apache2.....	25
6.2 ab: Apache2 ja Varnish.....	26
6.3 ab: nginx.....	27
7 Pohdinta.....	28
7.1 Tulokset.....	28
7.2 Tutkimuksen luotettavuus ja uskottavuus.....	29
7.3 Johtopäätökset ja ehdotukset.....	29

7.4 Opinnäytetyöprosessi.....	30
7.5 Oma oppiminen.....	31
Lähteet.....	33
Liitteet.....	37
Liite 1. Esimerkki Apache Benchmark -ohjelman tuloksista.....	37

# 1 Johdanto

Raspberry Pi 2 (jatkossa RPI2) on pienikokoinen yhden piirilevyn tietokone, jonka kehittäjä on brittiläinen hyväntekeväisyysjärjestö Raspberry Pi Foundation. Se on suunniteltu erityisesti ohjelmoinnin opetuskäyttöön, mutta soveltuu ominaisuuksiltaan monenlaiseen harrastus- ja viihdekäyttöön. (Raspberrypi a.) Vuoden 2015 helmikuussa julkaistussa Pi 2 Model B -versiossa on neliytiminen suoritin ja 1 gigatavu keskusmuistia (Raspberrypi b), mikä on laitteen 35 dollarin (Suomessa noin 50€) hinnan huomioon ottaen tehokas yhdistelmä.

Tämän lisäksi RPI2:lle on saatavilla käyttöjärjestelmäksi Raspian-niminen Linux-jakelu, joka perustuu Debian-käyttöjärjestelmään. (Raspbian a.) Kyseiselle edulliselle tietokoneelle on siis saatavilla ilmainen käyttöjärjestelmä, jolle on saatavilla palvelinkäytössäkin suositun Debianin sovelluksista suuri osa. (Raspbian b.)

Soveltuuko edellä kuvatun kaltainen edullinen tietokone palvelinkäyttöön? Tässä opinnäytetyössä tarkastellaan RPI2 ja Raspbian -yhdistelmää sovellusvalikoiman, fyysisten rajoitusten ja suorituskyvyn kannalta.

Tietoperustaosuudessa tutkitaan RPI2:n fyysisiä osia ja niiden vaikutusta ohjelmistojen valintaan ja toimintaan, kiintolevynä toimivien microSD-korttien soveltuvuutta tarkoitukseen sekä sitä, minkä kaltaisten sovellusten ajamiseen kone voi ylipäättään soveltua. Tutkittaviksi valitut palvelinsovellustyypit (DHCP, HTTP- ja nimipalvelin) ja niitä edustavat sovellukset esitellään.

Empiirisessä osuudessa kuvataan RPI2-tietokoneen ja käyttöjärjestelmän käyttöönotto ja asennetaan valitut palvelinsovellukset, joille tehdään testauksen vaatimat määrittymiset. Lisäksi kuvataan pystytetyn järjestelmän varmuuskopiointi ja kopion palauttaminen. Konetta käytetään etäyhteydellä SSH-protokollan avulla.

Lopuksi tarkastellaan asennettujen ohjelmistojen toimivuutta ja RPI2:n soveltumista palvelinkäyttöön toteutettujen testien valossa, joihin lukeutuu DHCP- ja nimipalvelinten toiminta testiympäristössä ja HTTP-palvelinten rasitustestaus staattisilla ja dynaamisilla sivuilla erilaisilla yhtäaikaisten yhteyksien määrillä.

Työssä käytettyjen termien selityksiä:

*ab*

Apache Benchmark. Verkkosivupalvelinten rasiustestaukseen soveltuva ohjelma.

*ARM*

Vähävirtainen, erityisesti mobiililaitteissa käytetty suoritinyyppi.

*DNS*

Domain Name Service. Internetin nimipalvelu, joka muuttaa numeeriset osoitteet ihmisten luettavaan muotoon ja toisin päin.

*DHCP*

Dynamic Host Configuration Protocol. Protokolla, jolla verkon tietokoneet voivat pyytää IP-osoiteasetuksensa palvelimelta.

*Dynaaminen sivu*

Verkkosivu, jolla on dynaamista sisältöä. Esimerkiksi sivu, joka luodaan skriptien ja tietokannasta haettavan datan pohjalta.

*IP-osoite*

Numeerinen osoite, jolla tietokone löytyy verkosta.

*KeepAlive*

Menetelmä, jolla avattua TCP-yhteyttä käytetään uudelleen HTTP-pyyntöjä ja -vastauksia varten.

*Paketinhallinta*

Käyttöjärjestelmän osa, jonka kautta voidaan asentaa ja poistaa kyseiselle käyttöjärjestelmälle valmiiksi paketoituja sovelluksia.

*Palvelin*

Verkkoon liitetty tietokone, jolla on toiminnassa sisääntulevia pyyntöjä kuunteleva palvelinsovellus (tai useampi). Voi myös tarkoittaa palvelinsovellusta.

*Staatinen sivu*

Verkkosivu, joka muodostetaan pelkän HTML-tiedoston pohjalta, ilman tarvetta skripteille tai tietokantayhteydelle



## 2 Työn tavoitteet ja raja

Tässä työssä tutkitaan laadullisin menetelmin Raspberry Pi 2 Model B -tietokoneen ja sille suunnitellun Raspbian-käyttöjärjestelmän soveltuvuutta palvelinkäyttöön. Tätä tutkitaan muutamasta näkökulmasta. Tutkimuskysymyksen voisi muotoilla seuraavasti:

- Soveltuuko Raspberry Pi 2 palvelinkäyttöön?

Kysymykseen haetaan vastausta seuraavien kysymysten avulla:

- Minkälaiseen palvelinkäyttöön laite voi soveltua?
  - Mihin sen suorituskyky riittää?
- Mitä palvelinsovelluksia on saatavilla?
- Kuinka ne toimivat käytännössä?

Työn teoriaosuudessa tarkastellaan tutkittavan tietokoneen olennaisia komponentteja sekä niiden asettamia rajoituksia. Työhön valitut palvelinsovellustyypit ja niihin liittyvät käsitteet määritellään, minkä jälkeen esitellään eri sovellusvaihtoehtoja. Koska erilaisten palvelinsovellusten kirjo Raspbianin paketinhallinnassa on valtava, keskitytään tässä työssä muutamaa yleisesti käytettyyn sovellukseen ja niiden kevyempiin vastineisiin.

Empiirisessä osuudessa kuvataan verkko, johon tietokone liitetään, käyttöjärjestelmän ja sovellusten asennus ja niiden käyttöönotto. Sen jälkeen asennettujen palvelinsovellusten toimintaa testataan käytännössä. Sovellusvalikoiman ja ohjetiedostojen ajantasaisuutta testataan asentamalla ja määrittelemällä asetukset valituille DHCP- ja nimipalvelinsovelluksille. RPI2:n suorituskykyä ja soveltuvuutta HTTP-palvelimeksi selvitetään rasiustestaamalla valittuja HTTP-palvelinsovelluksia pyytämällä rasiustestausohjelmalla staattisia ja dynaamisia sivuja erilaisilla yhtäaikaisten yhteyksien määrillä. Tällä on myös tarkoitus selvittää, paljonko HTTP-palvelinsovelluksen valinta vaikuttaa RPI2:n toimintaan.

Tässä työssä ei käsitellä tietoturvasäikeitä, eikä laitetta ja sovelluksia testata suurikokoisessa verkossa, sillä laite tuskin soveltuu suurikokoisen yrityksen käyttöön eikä tämän työn puitteissa ole mahdollista tällaista testiympäristöä rakentaa. Pienessä verkossa voidaan kuitenkin arvioida koneen toimintaa DHCP- ja nimipalvelukäytössä, ja rasiustestausohjelmalla voidaan simuloida useita yhtäaikaista yhteyksiä.

### 3 Tutkittava tietokone

Tässä luvussa kuvataan opinnäytetyön kohteena oleva tietokone, Raspberry Pi 2 Model B.

#### 3.1 Tietokoneen yleiskatsaus

RPI2:ssa on suorittimena 900-megahertsinen neliytiminen ARM Cortex-A7. Keskusmuistia on 1 gigatavu. Liitäntöistä olennaisimmat ovat 4 USB-porttia, HDMI-portti ja Ethernet-portti. (Raspberry pi c.) Lisäksi koneessa on 40 kappaletta GPIO-pinnejä (General purpose input-output), jotka ovat vapaasti käytettävissä ja ohjelmoitavissa, käyttötarkoitusten vaihdellessa LED-valoista hälyttimiin tai robotteihin. (Raspberrypi d.)

Koneen suoritin on ARMv7-tyyppinen, toisin kuin Raspberry Pi 1:ssä, jossa oli ARMv6-tyyppinen suoritin. Tämän takia RPI2:lla toimii laajempi kirjo ARM-pohjaisia Linux-jakeluita. Microsoftin Windows 10 -käyttöjärjestelmästä on myös ARMv7-suorittimella toimiva versio. (Raspberrypi c.)

Suorittimen, sen välimuistin ja keskusmuistin kellotaajuuksia voi nostaa koneen suorituskyvyn parantamiseksi. Muutokset eivät vaikuta takuuseen, mikäli tiettyjä jännitteeseen liittyviä asetuksia ei muuteta. (James 2015.) Raspbianin konfigurointityökalu tarjoaa valikoiman valmiita ylikellotusasetuksia.

RPI2 on suunnilleen luottokortin kokoinen (Kuva 1), joten sen fyysiset tilavaatimukset eivät yleensä aiheuta ongelmia.



Kuva 1: Raspberry Pi 2 Model B on pienikokoinen

### 3.2 Suoritin

RPI2:ssa on yksinkertaistetun käskykannan (RISC, Reduced Instruction Set Computer) ARM-suoritin. Tämänkaltaiset suorittimet poikkeavat pöytä- ja palvelinkoneissa yleisemmistä CISC-suorittimista (Complex Instruction Set Computer): suoritin ymmärtää vain muutamia yksinkertaisia käskyjä, joita ketjutetaan peräkkäin, siinä missä CISC-tyyppinen suoritin ymmärtää suuren määrän monimutkaisia käskyjä. RISC-tyyppiset käskyt ovat nopeampia suorittaa. (Thompson & Thompson 2003, 154.)

RPI2:n virrankulutus on vähäinen verrattuna perinteisiin palvelinkoneisiin: CISC-tyyppinen palvelinkone vie vähimmillään noin 300W virtaa (Peters 2015), kun taas RPI2:n virrankulutus lisälaitteineen (langaton verkkosovitin yms.) jää alle 5W:n. (RasPi 2014.)

ARM-suorittimia käytetään suurimassa osassa älypuhelimia ja mobiililaitteita; noin 99% mobiililaitteista on käytössä ARM:in suunnittelema suoritin. (Vance 2014.)

### 3.3 Tiedon tallennus

RPI2:ssa ei ole kiintolevyä eikä SATA/IDE-väylää sellaista varten. Kiintolevyn asemesta käyttöjärjestelmä ja käyttäjän tiedostot tallennetaan microSD-kortille. (Raspberrypi c.) Tämä vaikuttaa suorituskykyyn, sillä SD-kortit ovat luku- ja kirjoitusnopeuksiltaan vaatimattomia, jääden parhaimmillaan samoihin nopeuksiin kuin mekaaniset kiintolevyt ja reilusti nykyaikaisista SSD-kiintolevyistä jälkeen. (Jackson 2008.)

Flash-tyyppisissä tallennusmedioissa kirjoituskerrat ovat myös rajoitettuja, joten kortin kestävyys saattaa muodostua ongelmaksi. Tätä varten joissakin korteissa käytetään nykyään nk. ”wear leveling” -tekniikkaa, jossa tietoa tallennetaan hajautetusti ympäri korttia, jolloin kortti kuluu tasaisemmin. (Crump 2015.)

Kortin kestävyys on yleensä arkikäytössä hyväksyttävä. SD-kortin elinikä kirjoituskertoina liikkuu miljoonissa: mikäli kortin tallennuskapasiteettia täytetään kirjoittamalla maksiminopeudella jatkuvasti, kestää kortti silti useita vuosia, ellei se ole viallinen. (Jackson 2008.)

SD-kortin korvaaminen uudella ei ole merkittävä kuluera: 8 gigatavun kokoinen microSD-kortti maksaa edullisimmillaan noin kymmenen euroa. (Hintaseuranta 2015.) Toisaalta hinta per gigatavu on huomattavasti suurempi kuin perinteisissä kiintolevyissä. USB-liitännällinen ulkoinen kiintolevy voi olla yksi ratkaisu tähän. Laitteesta on myös olemassa Banana Pi -niminen versio, jossa on SATA-väylä kiintolevyä varten. (Bananapi 2014.)

## 4 Sovellukset

Tässä luvussa esitellään RPI2:lle suunniteltu Raspbian-käyttöjärjestelmä ja esitellään tutkittavat palvelinsovellustyypit ja sovellukset.

### 4.1 Raspbian-käyttöjärjestelmä

Raspbian on epävirallinen porttaus Debian-nimisen Linux-jakelun ARM-suorittimille tehdystä armhf-versiosta. Sen nykyversio hyödyntää RPI2:n ARMv7-suoritinta tehokkaammin kuin nk. armel-versio, jonka ARMv6-suorittimet (jollainen on mm. Raspberry Pi 1 -tietokoneessa) vaativat toimiakseen. (Raspbian b.)

Debian on alun perin vuonna 1993 julkaistu Linux-jakelu, johon muun muassa pöytä- ja palvelinkoneissa suosittu Ubuntu perustuu. (Ubuntu 2015.) Raspbianille on saatavilla suurin osa Debianille paketoituista sovelluksista (Raspbian b). Tämän takia ohjeita ja tukea on yleensä helposti löydettävissä, vaikka Raspbian itse on edellä mainittuja vähemmän käytetty käyttöjärjestelmä.

Raspbian seuraa Debiania viiveellä. Debianin versio 8, nimimerkiltä ”jessie” julkaistiin huhtikuussa 2015 (Debian 2015). Raspbianin vastaava versiopäivitys julkaistiin syyskuussa (Raspberrypi 2015). Toisaalta Debianin tietoturvapäivitykset pyritään tuomaan Raspbianille 12 tunnin sisällä niiden julkaisusta. (Raspberrypi 2013.)

Raspbian on siirtynyt Debianin myötä käyttämään systemd-nimistä ohjelmien käynnistysjärjestelmää (Armstrong 2014) syyskuussa 2015, minkä takia internetistä löytyvät sitä koskevat ohjeet ja resurssit eivät ole välttämättä vielä ajantasaisia.

Käyttöjärjestelmä on suunniteltu toimimaan sekä graafisessa työpöytäympäristössä että ilman, jolloin sitä hallitaan verkon yli. Koneen keskusmuistia ei siten tarvitse käyttää graafiseen työpöytäympäristöön, kun sitä käytetään palvelimena. Valintaa voidaan muuttaa raspi-config-työkalulla, joka on kuvattu luvussa 5.4.3.

### 4.2 Palvelinsovellukset

Tässä luvussa esitellään tutkittavaksi valitut protokollat DHCP, HTTP ja nimipalvelu (DNS) sekä tutkittaviksi valitut palvelinsovellukset.

### 4.2.1 DHCP

Verkon tietokoneilla on oltava IP-osoitteet (Internet protocol), jotta ne voivat kommunikoida toistensa kanssa. Osoitteet voidaan määrittellä manuaalisesti, jolloin puhutaan staattisesta konfiguraatiosta, mutta esimerkiksi suurikokoisissa yritysverkoissa tämä on työlästä ja aikaavievää. (Linuxconfig.)

DHCP-palvelin (Dynamic host configuration protocol) huolehtii IP-osoiteasetusten jakamisesta verkossa oleville koneille, kun niiden DHCP-asiakasohjelma niitä pyytää. Jaettu osoite on voimassa määritellyn ajan. Mikäli osoitteen saanut kone ei uusi lainaa tuon ajan jälkeen, osoite vapautuu uudelleen käytettäväksi. DHCP-palvelin tunnistaa koneet niiden laiteosoitteen perusteella (MAC address, media access control address), minkä takia tietty osoite voidaan sijoittaa tiettyyn laitteeseen. (Linuxconfig.)

DHCP:n ollessa käytössä asiakaskoneiden osoitemääritys helpottuu ja nopeutuu, minkä lisäksi IP-osoiteristiriitoja ei pääse syntymään. (Linuxconfig.)

Monissa nykyisissä aktiivilaitteissa on reitittimen, kytkimen yms. lisäksi DHCP-palvelin (Aakala 2009, 42), mutta erillinen DHCP-palvelinohjelmisto mahdollistaa yksityiskohtaisemman konfiguraation, mikä voi olla suurikokoisissa yritysverkoissa tarpeellista.

Raspbianin pakettihallinnasta löytyvistä DHCP-palvelinsovelluksista tässä työssä tarkasteltaviksi on valittu Internet Systems Consortiumin kehittämä ISC dhcp ja Simon Kelleyn kehittämä dnsmasq, joka toimii myös nimipalvelimena. ISC dhcp on hyvin käytetty, suuremmille verkoille soveltuva DHCP-palvelin (ISC 2015a), kun taas dnsmasq on suunniteltu pienempiä verkkoja varten (TheKelleys). Kumpikin palvelinsovellus on avointa lähdekoodia.

### 4.2.2 Nimipalvelin

Nimipalvelu (DNS, domain name service) huolehtii ihmisten luettavien koneiden nimien muuttamisesta internet-osoitteiksi. Kun käyttäjä kirjoittaa selaimeen osoitteen, esimerkiksi <http://www.example.com>, ottaa tietokone yhteyden nimipalvelimeen, joka etsii nimeä vastaavan IP-osoitteen. Yleensä tarkistetaan ensin koneen DNS-välimuisti, mikäli se on käytössä. Mikäli tietoa ei löydy välimuistista, kysytään osoitetta rekursiiviselta nimipalvelimelta (tällaisia ovat mm. palveluntarjoajan nimipalvelu, samoin Googlen DNS ja OpenDNS), jotka tarvittaessa vievät kyselyn eteenpäin. (Gonyea 2010.)

Nimipalvelimia on pääpiirteissään kahdenlaisia: autoritäärisiä nimipalvelimia ja resolvereita. Autoritäärinen nimipalvelin vastaa vain oman vyöhykkeeseensä liittyviin nimikyselyihin, ei rekursiivisiin kyselyihin, eikä käytä nimivälimuistia. Resolveri vastaa rekursiivisiin kyselyihin; sillä on pääsy koko maailman julkisiin nimitietoihin. Se käyttää nimivälimuistia liikenteen vähentämiseksi ja suorituskyvyn parantamiseksi. (Ellingwood 2014.)

Tässä työssä ei käsitellä autoritäärisiä nimipalvelimia.

Raspbianin paketinhallinnasta löytyvistä nimipalvelimista tässä työssä tarkasteltaviksi on valittu Internet Systems Consortiumin kehittämä BIND9 ja Simon Kelleyn dnsmasq, joka toimii myös DHCP-palvelimena. BIND9 on internetin eniten käytetty nimipalvelin (ISC 2015b). Dnsmasq on pienemmille verkoille suunniteltu, kevyt palvelinsovellus (TheKelleys). Kumpikin on avoimen lähdekoodin sovellus.

Nimihakujen tekemiseen löytyy Debianin paketinhallinnan dnsutils-nimisen metapaketin työkaluista dig- ja nslookup-nimiset ohjelmat. (Windows-käyttöjärjestelmässä voidaan käyttää käyttöjärjestelmästä löytyvää nslookup-komentoa.)

### **4.2.3 HTTP-palvelin**

HTTP-palvelin tarjoilee tiedostoja, usein verkkosivuja, HTTP-protokollaa käyttäen. Lyhenne HTTP tulee sanoista "Hypertext Transfer Protocol". Yleisesti käytettyjä palvelinsovelluksia ovat mm. Apache, Microsoftin IIS (Internet Information Server) ja nginx. (WhatIs 2015.)

Verkkosivuja on staattisia ja dynaamisia. Staattiset verkkosivut sisältävät vain HTML-kuvauskieltä ilman skriptejä tai tietokantayhteyttä (TechTerms 2009a). Dynaamisilla verkkosivuilla palvelin luo tarjottavan sivun reaaliajassa, yleensä sivulla käytettyjen skriptien ja mahdollisesti tietokannasta tulevan datan pohjalta. (TechTerms 2009b.)

Staattiset verkkosivut ovat palvelimelle nopeampia tarjoilla, mutta dynaamiset verkkosivut ovat monesti helpompia ylläpitää. Dynaamisella sivulla palvelimen täytyy luoda tarjottava sivu aina uudelleen, kun taas staattisen sivun tapauksessa tarjoilee se vain muuttumattomia tiedostoja, yleensä HTML- ja kuvatiedostoja. (Nicholson 2014.)

Välimuistina toimiva käänteinen välityspalvelin, kuten Varnish, (caching reverse proxy) voi keventää palvelimen raskuudesta dynaamisia verkkosivuja tarjottaessa, jolloin HTTP-liikennettä kuuntelee varsinaisen HTTP-palvelimen asemesta välityspalvelin. Tällöin esimerkiksi dynaamisen sisällönhallintajärjestelmän useimmiten pyydytetyt sivut luodaan

välityspalvelimen niitä pyytäessä, joka sitten tarjoilee ne välimuististaan eteenpäin, jolloin kuormittavia skriptejä ja tietokantayhteyksiä ei tarvita jokaista sivun näyttöä varten. (Winkler 2012.)

Koska RPI2 on suorituskyvyltään heikompi kuin varsinaiset palvelinkoneet, kiinnitetään tässä työssä huomiota edellä mainitun tyyppisen välityspalvelimen vaikutukseen kuormituksen vähentämiseksi.

Tässä työssä tarkasteltaviksi HTTP-palvelimiksi on valittu Apache HTTP Server ja nginx, sekä välityspalvelin Varnish. HTTP-palvelimen rasitustestaamiseen soveltuu ab-niminen (Apache HTTP server benchmarking tool) ohjelma, jolla voidaan lähettää palvelimelle haluttu määrä pyyntöjä tietyssä ajassa, jonka jälkeen se näyttää yksityiskohtaista tietoa sivunäyttöjen määrästä ja viiveistä. (Apache.)

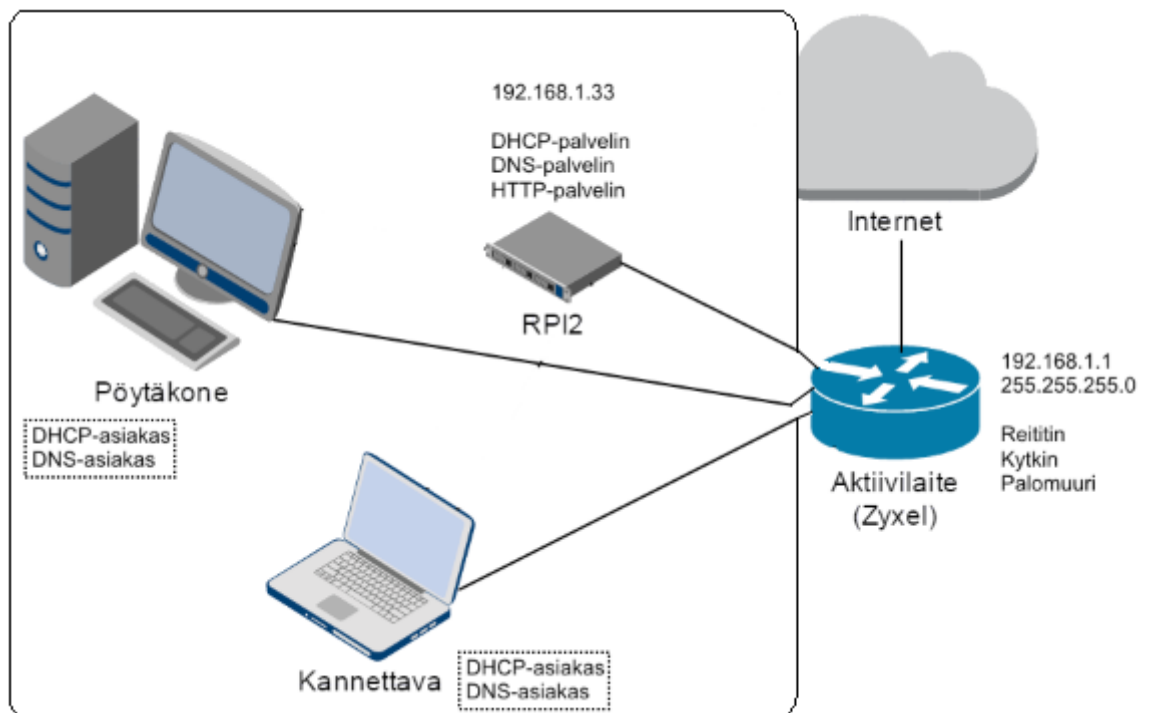
## 5 Sovellusten asennus ja ympäristö

Tässä luvussa kuvataan testausympäristö sekä RPI2:n, Raspbianin ja edellisessä luvussa esiteltujen palvelinsovellusten käyttöönotto ja testaus. Tarkoituksena on selvittää, kuinka sovellusten asennus paketinhallinnasta ja määritysten tekeminen onnistuu, sekä testata niiden toimivuutta käytännössä.

RPI2 liitetään lähiverkkoon, jossa se toimii DHCP- ja nimipalvelimena. Asennettujen HTTP-palvelinsovellusten suorituskykyä RPI2:lla mitataan luvussa 4.2.3 mainitulla ab-sovelluksella.

### 5.1 Testausympäristön kuvaus

RPI2 liitettiin lähiverkkoon Zyxel-merkkisen aktiivilaitteen taakse, jossa on reititin, kytkin, palomuuuri ja DHCP-palvelin, joka poistettiin käytöstä tämän työn ajaksi. Lähiverkossa on RPI2:n lisäksi yksi pöytäkone (konenimi: z97c) ja kannettava (konenimi: elitebook). Pöytäkone ja kannettava asetettiin pyytämään IP-osoiteasetukset DHCP-protokollaa käyttäen testausta varten (kuvassa 2).



Kuva 2: Testiympäristön laitteet verkossa



## 5.2 RPI2:n fyysinen kokoonpano

RPI2 sisältää vain luvussa 2.1 kuvatun yhden piirilevyn tietokoneen. Kiintolevynä toimiva microSD-kortti, muistikortinlukija, virtalähde (USB-laturi) ja tarvittavat kaapelit (Ethernet, USB-virta) hankittiin erikseen, mikä vaikutti käyttöönoton kustannuksiin muutamalla kymmenellä eurolla. Työssä käytetyt muistikortit ovat Transcend-merkkisiä 8GB Class 10 microSDHC-kortteja.

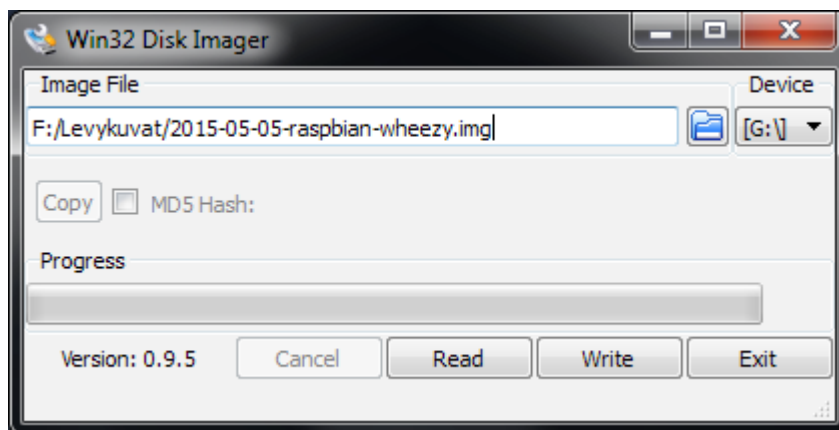
## 5.3 Muistikortille asennus ja varmuuskopiointi

Kiintolevynä toimivan microSD-kortin käyttöjärjestelmäasennus ja varmuuskopiointi kuvataan tässä luvussa.

### 5.3.1 Levykuvan tiputus muistikortille

Raspbian-käyttöjärjestelmän levykuvan saa ladattua Raspberry Pi Foundationin verkkosivuilta osoitteesta <https://www.raspberrypi.org/downloads/raspbian>. Latauksen eheyden tarkistamiseksi sivulla on näkyvillä levykuvatiedoston SHA-1-tarkistusluku.

Windows-käyttöjärjestelmässä levykuvan kirjoittamiseen muistikortille laitevalmistaja suosittelee Win32 Disk Imager -nimistä avointa ohjelmaa ja Linux-käyttöjärjestelmissä niistä löytyvää dd-työkalua (Raspberrypi e).



Kuva 3: Levykuvan tiputus muistikortille Win32 Disk Imager -ohjelmassa

Ladattu levykuva tiputettiin muistikortille edellä mainitulla Win32 Disk Imager -ohjelmalla erillistä muistikortinlukijaa käyttäen Windows 7 -käyttöjärjestelmässä.

### **5.3.2 Asennetun järjestelmän varmuuskopiointi levykuvaksi**

Muistikortin sisällön voi myös varmuuskopioida levykuvaksi. Tässä työssä kopiointiin käytettiin luvussa 5.3.1 esiteltyä Win32 Disk Imager -ohjelmaa.

Koska muistikortti on irrotettava RPI2:sta ennen muistikortin sisällön kopiointia ja virtajohdon irrottaminen voi korruptoida muistikortin sisällön tai häiritä asennettua käyttöjärjestelmää, on laite ensin sammutettava. Koska RPI2:ssa ei ole virtakytkintä eikä graafinen työpöytä ole käytössä, käytettiin komentoa "shutdown" parametreilla -h 0, jolloin kone sammuttaa itsensä heti.

Kopiointi tehtiin kuten aiemmin kuvattu levykuvan tiputus, mutta Write-painikkeen (kirjoita) sijaan käytettiin Read-painiketta (lue), jolloin sisältö tallennetaan tekstikentässä määriteltyyn tiedostoon.

Työn aikana kävi ilmi, että muutokset muistikortin sisältöön toisella tietokoneella voivat estää RPI2:n käyttöjärjestelmän päätyksen toimimattomaan tilaan. Muistikortille kirjoittamisen voi estää microSD-SD-adapterilla. Tällainen adapteri toimitettiin työssä käytetyn muistikortin mukana, joten sitä käytettiin kopioitaessa.

## **5.4 Käyttöjärjestelmän käyttöönotto ja määrittelyt**

Muistikortille tiputetun Raspbian-käyttöjärjestelmän käyttöönotto, määrittelyt ja etähallinta kuvataan tässä luvussa.

### **5.4.1 Etähallinta SSH:lla**

Tässä työssä RPI2:ta hallittiin SSH-asiakasohjelmalla verkon yli. Tähän käytettiin Simon Tathamien kehittämän PuTTY-ohjelman Windows-versiota ja tarvittaessa Linux-jakelujen ssh-ohjelmaa.

Raspbian-levykuvan käyttöjärjestelmässä on SSH-palvelin oletuksena päällä kuuntelemassa porttia 22, minkä lisäksi se pyytää osoiteasetuksia DHCP:llä (Instructables 2015). Tässä työssä laitteen saama IP-osoite saatiin selville kotireitittimen DHCP-palvelimen hallintapaneelista. Saatu osoite oli 192.168.1.100/24 (/24 tarkoittaa samaa kuin aliverkon peite 255.255.255.0).

Raspbianissa on oletuksena käyttäjä "pi", jonka on mahdollista suorittaa komenoja pääkäyttäjän oikeuksilla. Tässä työssä pystytetty ympäristö ei näy internetiin, joten oletuskäyttäjätunnusta käytettiin kaikissa asennustoimenpiteissä.

Näiden tietojen pohjalta RPI2:lle kirjaututtiin avaamalla SSH-yhteys osoitteeseen 192.168.1.100, porttiin 22, käyttäjätunnuksella "pi" ja oletussalasanalla "raspberry", joka vaihdettiin ensikirjautumisen jälkeen komennolla "passwd".

#### 5.4.2 Staattinen IP-osoite

RPI2:lle haluttiin staattinen IP-osoite, jotta se ei ole riippuvainen kotireitittimen DHCP-palvelusta ja vastaa aina samasta osoitteesta.

Modmypi-verkkosivun (2013) ohjeen avulla muokattiin tiedostoa /etc/network/interfaces, jossa määriteltiin RPI2:n verkkokortille interface eth0, jolla on staattinen ip-osoite 192.168.1.33. Kyseisen tiedoston sisältö:

```
auto lo
iface lo inet loopback

auto eth0
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.33
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 8.8.8.8 8.8.4.4
```

Raspbianin DHCP-asiakasohjelma, dhcpcd5, poistettiin myös, jotta ylläoleva staattisen IP-osoitteen määrittely toimisi; tämä muutos tehtiin myös, jottei edellä mainittu dhcpcd5 häiritse ISC DHCP serverin tai dnsmasq:n toimintaa.

#### 5.4.3 Muut asetukset ja raspi-config

Raspbianissa on järjestelmän perusasetusten määrittämistä varten työkalu nimeltä raspi-config. Sillä voidaan muuttaa muun muassa oletuskäyttäjän salasana, kieli- ja aikavyöhykeasetukset, laajentaa tiedostojärjestelmä kattamaan muistikortin koko tallennuskapasiteetti tai toteuttaa luvussa 3.1 mainittu ylikellotus. Ylikellotusasetukseksi valittiin valmiista vaihtoehdoista "Pi2 1000MHz ARM, 500MHz core, 500MHz SDRAM, 2 overvolt".

Työkalulla muutettiin RPI2:n konenimeksi "ont", määriteltiin oikea aikavyöhyke, minkä lisäksi toteutettiin edellä mainittu tiedostojärjestelmän laajennus sekä ylikellotus.

## 5.5 Palvelinsovellukset

Luvussa 4.2 mainittujen palvelinsovellusten asennus ja määrittelyt käsitellään tässä luvussa. Eri testauskokoonpanojen pohjana toimi käyttöjärjestelmästä tehty levykuva, jossa oli tehty lukujen 5.4.2 ja 5.4.3 kuvatut toimenpiteet. Näin ollen voitiin testata dnsmasq:n DHCP- ja nimipalvelin yhtenä kokonaisuutena, ISC DHCP ja BIND9 toisena ja myös eri HTTP-palvelimet omina kokoonpanoinaan.

### 5.5.1 dnsmasq

Dnsmasq on DHCP- ja nimipalvelin. Koska RPI2:lle annettiin staattinen IP-osoite ja Raspbianin oletuksena toimiva dhcpd5 poistettiin käytöstä, ei dnsmasq:n asennus vaatinut muita valmisteluja. Palvelinsovellus asennettiin paketinhallinnasta komennolla:

```
sudo apt-get install dnsmasq
```

Dnsmasq toimi välittömästi asennuksen jälkeen caching-tyyppisenä nimipalvelimena, välittäen nimikyselyt /etc/network/interfaces-tiedostossa määritellyille Googlen nimipalvelimille (8.8.8.8 ja 8.8.4.4). Jotta myös DHCP-palvelin saatiin käyntiin, muokattiin dnsmasq:n asetustiedostoa /etc/dnsmasq.conf. Sen lopulliseksi sisällöksi tuli:

```
domain-needed  
bogus-priv  
no-resolv  
server=8.8.8.8  
server=8.8.4.4  
dhcp-range=192.168.1.100,192.168.1.130,255.255.255.0,12h  
dhcp-option=option:router,192.168.1.1
```

Kyseinen asetustiedosto on hyvin kommentoitu, ja käyttöönotossa pitäisi yleensä riittää kommenttimerkin (#) poisto muutaman rivin edestä. Lähiverkon tietokoneet eivät päässeet internetiin ilman aliverkon peitteen ja oletusyhdyskäytävän määrittelyä, mutta näkivät toisensa. Asetusten määrittelyssä käytettiin apuna Yi Qing Sim -blogia (Yiqingsim 2014). Palvelinsovellus käynnistettiin muutosten jälkeen uudelleen komennolla "sudo service dnsmasq restart".

Lähiverkon pöytäkone (z97c) ja kannettava (elitebook) määriteltiin tämän jälkeen DHCP-asiakkaiksi. Kumpikin kone sai dnsmasq:lta osoitteen; tämä varmistettiin tiedostosta /var/lib/misc/dnsmasq.leases:

```
pi@ont ~ $ cat /var/lib/misc/dnsmasq.leases
1446775521 00:27:10:68:e9:28 192.168.1.125 elitebook *
1446775493 78:24:af:bd:3e:26 192.168.1.111 z97c      01:78:24:af:bd:3e:26
```

Nimipalvelun toiminta testattiin pöytäkoneella Windowsin nslookup-komennolla:

```
C:\Users\lantero>nslookup www.haaga-helia.fi
Server: UnKnown
Address: 192.168.1.33
```

```
Non-authoritative answer:
Name:   www.haaga-helia.fi
Address: 37.233.91.219
```

Kannettavalla tietokoneella käytettiin Linux Mint -jakelussa dig-ohjelmaa komennolla "dig [www.haaga-helia.fi](http://www.haaga-helia.fi)", jonka SERVER-riviltä nähtiin käytetty nimipalvelin:

```
:: SERVER: 192.168.1.33#53(192.168.1.33)
```

## 5.5.2 BIND9

BIND9-nimipalvelimen käyttöönotossa käytettiin apuna Digitalocean-palveluntarjoajan tutoriaalia, jossa kuvataan asetustiedostoon vaadittavia kohtia (Ellingwood 2014b).

Kuten dnsmasq, myös BIND9 vaatii koneella olevan staattisen IP-osoitteen toimiakseen suoraan asennuksen jälkeen. BIND9 asennettiin Raspbianin paketinhallinnasta komennolla:

```
sudo apt-get install bind9 bind9utils bind9-doc
```

Palvelimen asetustiedosto on /etc/bind/named.conf.options. BIND toimi caching-tyyppisenä nimipalvelimena ilman muutoksia asetustiedostoon. Toiminta testattiin Windowsin nslookup-komennolla kuten luvussa 5.5.1:

```
C:\Users\lantero>nslookup www.eduskunta.fi
```

*Server: UnKnown*  
*Address: 192.168.1.33*

*Non-authoritative answer:*  
*Name: www.eduskunta.fi*  
*Address: 217.71.145.71*

BIND9 saadaan toimimaan forwarding-tyyppisenä nimipalvelimena lisäämällä asetustiedostoon seuraavat kursiivittomalla tekstillä merkityt rivit (Ellingwood 2014b):

```
options {  
directory "/var/cache/bind";  
  
forwarders {  
8.8.8.8;  
8.8.4.4;  
};  
forward only;  
  
dnssec-validation auto;  
  
auth-nxdomain no; # conform to RFC1035  
listen-on-v6 { any; };  
};
```

Tämä testattiin kannettavalla tietokoneella Linux Mintissä dnsutils-pakettiin kuuluvalla nslookup-komennolla:

```
antero@elitebook ~ $ nslookup www.eduskunta.fi  
Server: 192.168.1.33  
Address: 192.168.1.33#53  
  
Non-authoritative answer:  
Name: www.eduskunta.fi  
Address: 217.71.145.71
```

BIND9 toimi caching- ja forwarding-tyyppisenä nimipalvelimena vähäisellä asetusten muuttamisella, eikä nimihauissa lähiverkon koneilla ollut ongelmia.

### 5.5.3 ISC DHCP

ISC DHCP server asennettiin edellisten lukujen tapaan Raspbianin paketinhallinnasta apt-getillä:

```
sudo apt-get install isc-dhcp-server
```

Koska RPI2:lle annettiin aiemmin staattinen IP-osoite, ei asennuksessa tullut virheilmoituksia. Mikäli näin ei olisi tehty, ei palvelinsovellus käynnistyisi (Noveldevices).

ISC DHCP serverin perusasetukset löytyvät kahdesta tiedostosta: /etc/default/isc-dhcp-server, jossa sille kerrotaan, mistä varsinainen konfiguraatio löytyy, mikä prosessi-ID (pid) sille halutaan ja mitä interfacea se palvelee. Tiedostossa /etc/dhcp/dhcpd.conf määritellään DHCP:n osoiteavaruus ja palvelimet.

Testiympäristössä /etc/default/isc-dhcp-server-tiedoston sisältö oli:

```
DHCPD_CONF=/etc/dhcp/dhcpd.conf  
DHCPD_PID=/var/run/dhcpd.pid  
INTERFACES="eth0"
```

joka osoittaa DHCPD\_CONF-rivillä pääasetustiedoston ja INTERFACES-rivillä luvussa 5.4.2 tehdyn osoitemäärittelyn mukaisen eth0-interfacen kuunneltavaksi.

Asetustiedostossa /etc/dhcp/dhcpd.conf palvelimen DHCP-osoiteavaruus määriteltiin seuraavasti:

```
subnet 192.168.1.0 netmask 255.255.255.0 {  
range 192.168.1.100 192.168.1.130;  
option domain-name-servers 192.168.1.33;  
option routers 192.168.1.1;  
option broadcast-address 192.168.1.255;  
}
```

Pöytäkone (z97c, kuva 4) ja kannettava (elitebook) saivat osoiteasetukset onnistuneesti.

Tämä tarkistettiin ISC DHCP serverin luomasta tiedostosta /var/lib/dhcp/dhcpd.leases.

Merkintöjä on lyhennetty (merkintä "..."), koska kyseisen tiedoston merkinnät ovat pitkiä:

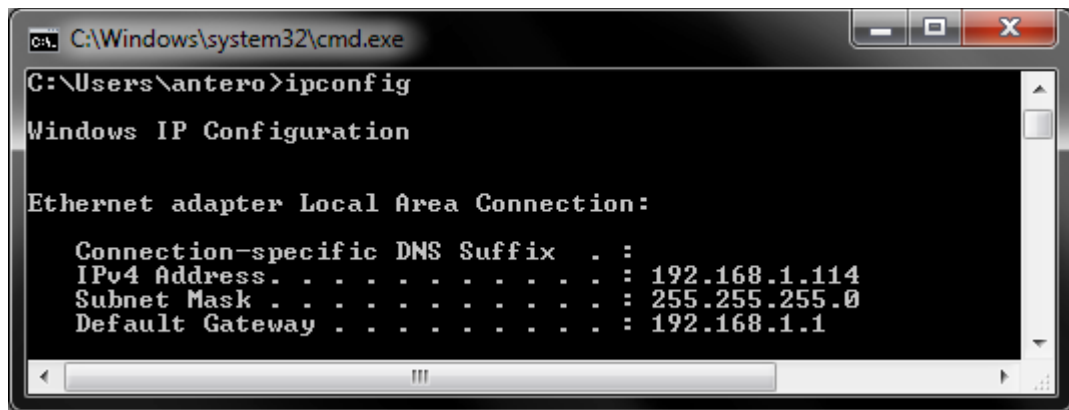
```
lease 192.168.1.114 {  
...  
hardware ethernet 78:24:af:bd:3e:26;  
uid "\001x$\257\275>&";
```

```

client-hostname "z97c";
}

lease 192.168.1.125 {
...
hardware ethernet 00:27:10:68:e9:28;
client-hostname "elitebook";
}

```



Kuva 4: Pöytäkoneen (z97c) saama IP-osoite

ISC DHCP server ei käynnistynyt Raspbianin jessie-versiossa uudelleenkäynnistyksen yhteydessä oikein, vaikka koneelle oli määritelty staattinen IP-osoite ja osoitettu oikea interface (eth0) kuunneltavaksi. Service status -komento palautti uudelleenkäynnistyksen jälkeen virheilmoituksen (ilmoitusta lyhennetty, merkintä "..."):

```

pi@ont ~ $ sudo service isc-dhcp-server status
...
Active: failed (Result: exit-code) since Wed 2015-11-04 14:44:08 EET; 51s ago
...
Nov 05 18:48:33 ont dhcpd[444]: Not configured to listen on any interfaces!
...
Nov 04 14:44:08 ont systemd[1]: Failed to start LSB: DHCP server.

```

Työn aikana ei saatu selville, mistä virhe johtuu. ISC DHCP server löytää tarvitsemansa asetustiedostot ja kuunneltavan interfacen kun käynnistyksen jälkeen annetaan service restart -komento:

```

pi@ont ~ $ sudo service isc-dhcp-server restart
pi@ont ~ $ sudo service isc-dhcp-server status
...

```



*Active: active (running) since Thu 2015-11-05 18:49:41 EET; 1s ago*

*...*

*Nov 05 18:49:41 ont isc-dhcp-server[877]: Starting ISC DHCP server: dhcpcd.*

*Nov 05 18:49:41 ont systemd[1]: Started LSB: DHCP server.*

#### 5.5.4 nginx

Nginx asennettiin Raspbianin paketinhallinnasta komennolla apt-get. Samalla asennettiin mysql-tietokantapalvelin ja php-kielen vaatimat paketit mysql- ja nginx-palvelimia varten:

```
sudo apt-get install nginx mysql-server php5 php5-fpm php5-fpm php5 mysql
```

Palvelin toimi heti asennuksen jälkeen. Lähiverkon muilla koneilla selaimella avautui nginxin oletussivu (/var/www/html/index.html) osoitteessa <http://192.168.1.33>. Koska myöhemmin testattiin staattisten sivujen lisäksi dynaamista sisältöä, muokattiin oletussivuston asetuksia siten, että php-kielen tuki on käytössä. Muokattava tiedosto oli /etc/nginx/sites-available/default, johon piti lisätä seuraavat rivit:

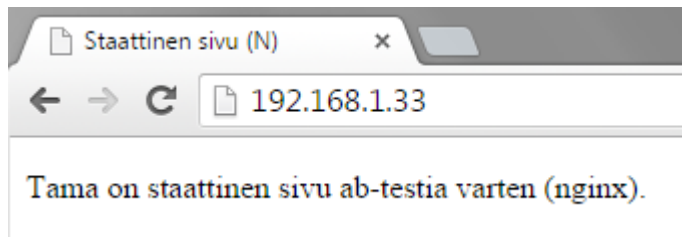
```
location ~ \.php$ {  
    fastcgi_pass unix:/var/run/php5-fpm.sock;  
    fastcgi_index index.php;  
    include fastcgi.conf;  
}
```

Raspbianin paketinhallinnasta löytyvän nginxin oletusasetustiedoston kommentti oli työn tekemisajankohtana väärin; include-riville ehdotetaan jatkoksi ”fastcgi\_params;”, mutta tällä asetuksella nginx ei ohjaa php-tiedostoja oikeaan paikkaan (Stackoverflow 2013), jolloin esimerkiksi /var/www/html/index.php ei toiminut.

Staattiseksi sivuksi luotiin yksinkertainen html5-standardin mukainen sivu, jossa näkyy käytetyn palvelimen nimi:

```
<!doctype html>  
<title>Staattinen sivu (N)</title>  
<p>Tama on staattinen sivu ab-testia varten (nginx).</p>
```

Palvelimen toiminta testattiin pöytäkoneella Google Chrome -selaimella (Kuva 5).



Kuva 5: Staattinen sivu avattuna selaimessa

### 5.5.5 Apache2

Apache2-palvelin asennettiin paketinhallinnasta samoin kuin nginx aiemmassa luvussa, minkä lisäksi tälläkin kertaa asennettiin myös mysql-server ja php-kielen vaatimat paketit.

```
sudo apt-get install apache2 mysql-server php5 php5-mysql
```

Staattinen sivu luotiin kuin luvussa 5.5.4, mutta tekstillä "Tämä on staattinen sivu ab-testia varten (apache)." Apache2 ei vaatinut staattisia sivuja varten muita asetuksia ja toimi siten suoraan asennuksen jälkeen.

### 5.5.6 Varnish ja Apache2

Luvussa 4.2.3 mainittu käänteinen välityspalvelin, Varnish, asetettiin toimimaan Apache2-palvelimen eteen. Edellisen luvun toimenpiteiden lisäksi asennettiin itse Varnish paketinhallinnasta:

```
sudo apt-get install varnish
```

Sen jälkeen Apache siirrettiin vastaamaan portista 8080, eikä portista 80, joka on HTTP-protokollan oletusportti. Muutokset tehtiin seuraaviin tiedostoihin:

```
/etc/apache2/ports.conf  
→ Listen 8080  
/etc/apache2/sites-available/000-default.conf  
→ <VirtualHost *:8080>
```

Varnish siirrettiin vastaamaan portista 80 ja hakemaan näytettävä sisältö edellä määritellystä portista 8080. Tätä varten muokattiin tiedostoja:

```
/etc/default/varnish
```

```
→ DAEMON_OPTS="-a :80 \  
/lib/systemd/system/varnish.service  
/etc/systemd/system/varnish.service  
→ ExecStart=/usr/sbin/varnishd -a :80 ...
```

Edellä mainitut tiedostomuutokset otettiin käyttöön ja varnish käynnistettiin uudelleen:

```
sudo systemctl daemon-reload  
sudo service varnish stop  
sudo service varnish start
```

Muutosten toimivuus testattiin pysäyttämällä Varnish ja kirjoittamalla pöytäkoneella Google Chrome -selaimen osoiteriville <http://192.168.1.33:80>, jolloin saatiin vastaukseksi "This webpage is not available ERR\_CONNECTION\_REFUSED". Kun Varnish käynnistettiin uudelleen ja kokeiltiin samaa osoitetta, saatiin auki luotu staattinen sivu ("Tämä on staattinen sivu ab-testia varten (apache+varnish)").

Tällä testillä varmistettiin, että asennettu Apachen ja Varnishin yhdistelmä toimii halutulla tavalla.

### 5.5.7 Wordpress

Wordpress-sisällönhallintajärjestelmän (jatkossa WP) asentaminen Apache- ja nginx-palvelinkonfiguraatioille kuvataan tässä kappaleessa, koska vaaditut toimenpiteet olivat samat kummankin palvelinsovelluksen kohdalla.

WP:n sisältävä paketti haettiin wget-ohjelmalla WP:n verkkosivulta hakemistoon /var/www/html ja purettiin sen alle hakemistoon /wordpress:

```
cd /var/www/html  
sudo wget http://wordpress.org/latest.tar.gz  
sudo tar -zxvf latest.tar.gz
```

WP vaatii mysql-tietokannan toimiakseen. Sitä varten luotiin tietokanta nimeltä "wordpress" seuraavilla komennoilla

```
mysql -uroot -ppassu  
create database wordpress;  
exit
```

WP:n toiminta testattiin avaamalla pöytäkoneella Google Chrome -selaimella osoite <http://192.168.1.33/wordpress>, jossa asennus saatettiin loppuun antamalla aiemmassa kappaleessa luodun tietokannan tiedot. WP:iin luotiin testausta varten uusi päivitys otsikolla "Ladattava sisältö", jonka sisällöksi annettiin "Tämä on ab-testia varten ladattava sivu" (Kuva 6).



Kuva 6: Wordpress-sivu valmiina rasiustestausta varten

Näin saatiin varmistettua, että tehdyt asennukset (Apache/nginx/varnish, mysql-server ja php5) toimivat käytännössä.

## 6 HTTP-palvelinten suorituskyvyn testaus

Tässä kappaleessa kuvataan RPI2:n suorituskyvyn testaus luvussa 5 asennettujen HTTP-palvelinsovellusten avulla. Asennetuilta HTTP-palvelimilta pyydettiin staattisia ja dynaamisia sivuja ab-nimisellä rasiustestausohjelmalla.

Ab-testausohjelmalla pyydettiin palvelimilta sivuja jatkuvasti puolen minuutin ajan vaihtelevilla yhtäaikaisten pyyntöjen määrillä (1, 5, 25 ja 50). Käytetty komento oli:

```
ab -t 30 -n 1000000 -c * -k -s 10000
```

jossa parametreilla määritellään seuraavat asetukset:

- t = aika sekunteina, määrittelee myös 50000 pyyntöä
- n = sivupyyntöjen määrä (mikäli edellä mainittu 50000 ehtii täytyä)
- c = yhtäaikaiset pyynnöt
- k = käytetään samaa yhteyttä (KeepAlive)
- s = aikakatkaisun viive sekunteina (jotta testi etenee varmasti loppuun)

RPI2:lla oli testien aikana auki top-niminen ohjelma (Kuva 7), jolla voidaan tarkkailla sen oman suorittimen ja keskusmuistin käyttöasteita. Se näyttää suorittimen käytön järjestelmä- ja käyttäjäkohtaisina prosenttilukuina, sekä keskusmuistin käytetyn ja vapaan määrän. Keskusmuistin käyttöaste laskettiin jakamalla käytetty määrä kokonaismäärällä.

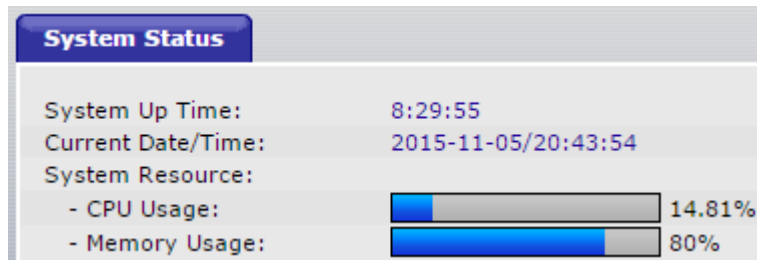
```
top - 19:02:15 up 3 min, 3 users, load average: 0.99, 0
Tasks: 118 total, 6 running, 112 sleeping, 0 stopped,
%Cpu(s): 94.7 us, 2.7 sy, 0.0 ni, 1.9 id, 0.0 wa, 0.
KiB Mem: 948120 total, 276396 used, 671724 free,
KiB Swap: 102396 total, 0 used, 102396 free.
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM
1300	www-data	20	0	113504	26756	15824	R	87.6	2.8
1303	www-data	20	0	107008	20672	15904	R	81.7	2.2
1305	www-data	20	0	106564	20204	15840	R	73.2	2.1
786	www-data	20	0	110380	23500	15832	R	67.3	2.5
787	www-data	20	0	112236	25640	15732	R	67.3	2.7
938	mysql	20	0	324008	42288	8880	S	9.5	4.5
571	www-data	20	0	15576	4236	2900	S	2.6	0.4

Kuva 7: Top-ohjelma seuraa resurssien käyttöä

Top löytyy valmiiksi monista Debian-pohjaisista käyttöjärjestelmistä. Samankaltainen resurssientarkailusovellus, atop, näyttäisi lisäksi verkkokortin ja kiintolevyn käyttöasteen (Koski 2010, 184). Atopin verkkokortin tarkkailu ei kuitenkaan toiminut käytetyssä Raspbianin versiossa. (Tätä näkymää pyydettyessä saatiin virheilmoitus "No kernel-patch installed, request ignored!")

Pöytäkoneella pidettiin auki lähiverkon aktiivilaitteen ("Zyxel" kuvassa 2) hallintapaneelia (Kuva 7), jotta voitiin tarkkailla sen prosessorin ja muistin käyttöasteita.



Kuva 8: Aktiivilaitteen hallintapaneelista nähdään prosessorin ja muistin käyttö

Mikäli aktiivilaitteen suorituskyky ei ole riittävä, eivät ab-testit anna luotettavaa dataa RPI2:n suorituskyvystä. Tässä tutkimuksessa toteutettujen testien aikana laitteen suorittimen käyttöaste pysyi noin 25%:n tasolla. Laitteen muistin käyttöaste nousi korkeimmillaan noin 90%:n tasolle.

Ab-testien tulokset esitetään taulukkoina, joissa on pylväinä:

- Palvelinsovellus: Apache2, varnish (Apache2:n edessä), nginx
- Yhteydet: Yhtäaikaiset sivupyyntöjen määrä
- Sisällön tyyppi: Pyydetäänkö staattista ("St.") vai dynaamista ("Dyn.") sivua
- Sivunäytöt: Kuinka monta sivua palvelinsovellus tarjoi määritellyssä ajassa
- Aika/näyttö (ms): Kuinka paljon aika yhteen sivupyyntöön meni (millisekunneissa)
- Näytöt/sekunti: Kuinka monta sivupyyntöä sovellus suoritti sekunnissa (keskiarvo)
- Suorittimen käyttöaste: Top-ohjelman näyttämä suorittimen käyttöaste testin aikana

- Keskusmuistin käyttöaste: Top-ohjelman tuloksista laskettu keskusmuistin käyttöaste testin aikana

Ab:n vaste on sisällöltään ja muotoilultaan pitkä. Esimerkki sen kokonaisesta tulosteesta on esitelty liitteessä 1.

## 6.1 ab: apache2

Tässä luvussa käsitellään Apache2- palvelinsovelluksen ab-testi ja sen tulokset.

Palvelin-sovellus	Yhteydet	Sisällön tyyppi	Sivunäytöt	Aika /näyttö (ms)	Näytöt /sekunti	Suorittimen käyttöaste	Keskusmuistin käyttöaste
Apache2	1	St.	22885	1	762,8	8,7%	50,6%
Apache2	5	St.	79503	2	2650,05	33,8%	50,9%
Apache2	25	St.	93527	8	3117,53	46,4%	55,3%
Apache2	50	St.	90073	17	3002,33	45,8%	59,0%
Apache2	1	Dyn.	72	421	2,37	24,3%	60,5%
Apache2	5	Dyn.	245	609	8,15	93,8%	60,9%
Apache2	25	Dyn.	229	3142	7,62	96,0%	68,1%
Apache2	50	Dyn.	201	6539	6,7	96,1%	76,2%

Taulukko 1: apache2-sovelluksen rasiustestin tulokset

Testeissä kävi ilmi, että RPI2:lla toimiva Apache2-palvelinsovellus selviää staattisten sivujen näyttämisestä 50:n yhtäaikaisen pyynnön määrällä melko hyvin: 17 millisekuntia ei ole ihmiskäyttäjälle vielä huomattava viive, ja sivuja palvelin ehti 30 sekunnin aikana tarjoilla noin 90 000. RPI2:n suorittimen ja keskusmuistin käyttöasteet jäivät staattisia sivuja tarjoillessa noin puoleen.

Dynaamisia sivuja tarjoillessa ero oli merkittävä. 50:n yhtäaikaisen yhteyden määrällä jäi sivunäyttöjen määrä noin kahteen sataan. Sivunäyttöjen määrä on vain kaksi promillea staattisten sivujen määrästä. Lisäksi pyyntöön vastaamisen viive kasvoi yli kuuteen sekuntiin. RPI2:n suorittimen käyttöaste oli jo viiden yhtäaikaisen sivupyynnön kohdalla lähellä maksimia, mutta keskusmuistin määrä ei ollut ongelma. Muistin käyttö alkoi kuitenkin nousta tuntuvasti yhtäaikaisten yhteyksien määrän kasvaessa. RPI2 ja Apache2 tuskin soveltuvat nykyaikaisten verkkosivujen palvelinkonfiguraatioksi.

## 6.2 ab: Apache2 ja Varnish

Tässä luvussa käsitellään Apache2- palvelinsovelluksen ja Varnish-välityspalvelimen yhdistelmän ab-testi ja sen tulokset.

Palvelin-sovellus	Yhteydet	Sisällön tyyppi	Sivunäytöt	Aika /näyttö (ms)	Näytöt /sekunti	Suorittimen käyttöaste	Keskusmuistin käyttöaste
varnish	1	St.	13617	2	453,88	7,3%	35,3%
varnish	5	St.	24864	6	814,56	16,7%	35,6%
varnish	25	St.	25107	28	812,8	22,5%	36,5%
varnish	50	St.	25133	54	817,3	5,8%	36,7%
varnish	1	Dyn.	8883	3	296,08	14,2%	39,2%
varnish	5	Dyn.	19231	8	641,02	16,4%	39,3%
varnish	25	Dyn.	22159	34	738,62	19,9%	39,6%
varnish	50	Dyn.	21849	69	728,3	22,1%	40,0%

Taulukko 2: apache2+varnish-sovelluksen rasiustestin tulokset

Kun Apache2:n eteen asetettiin Varnish-välityspalvelin, muuttuivat tulokset dynaamisten sivujen kohdalla merkittävästi: sivunäytöt nousivat suurimassa testattavassa kuormassa edellä mainitusta kahdesta sadasta yli kahteenkymmeneen tuhanteen, ja viive putosi yli kuudesta sekunnista alle sekunnin kymmenykseen. RPI2 saattaa soveltua tämänkaltaisen välityspalvelimen kanssa käytettynä HTTP-palvelimeksi, mutta tiedon tallennuksessa nousee silti ongelmaksi tiedon eheyden ja saatavuuden takaaminen, kiintolevynä toimivan microSD-kortin takia.

Staattisten sivujen kohdalla sivunäyttöjen määrä pieneni, mutta ab-ohjelmalla ei saatu Varnishin kohdalla yhdenmukaisia tuloksia. Sama testi staattisten sivujen kohdalla antoi sattumanvaraisesti hyvin vaihtelevia tuloksia. Esimerkiksi viidellä yhtäaikaisella yhteydellä sivunäyttöjen määräksi tuli muutaman kerran noin 100, vaikka pääasiassa tulos oli noin 25 000. Dynaamisia sivuja pyydetäessä tulokset olivat johdonmukaisia.

Syy ongelmaan ei selvinnyt työn aikana.



### 6.3 ab: nginx

Tässä luvussa käsitellään nginx-palvelinsovelluksen ab-testi ja sen tulokset.

Palvelinsovellus	Yhteydet	Sisällön tyyppi	Sivunäytöt	Aika /näyttö (ms)	Näytöt /sekunti	Suorittimen käyttöaste	Keskusmuistin käyttöaste
nginx	1	St.	28596	1	953,2	4,3%	24,5%
nginx	5	St.	122303	1	4076,75	16,8%	25,6%
nginx	25	St.	223038	3	7434,56	27,4%	27,1%
nginx	50	St.	221782	7	7392,68	32,3%	30,0%
nginx	1	Dyn.	77	392	2,55	29,3%	38,3%
nginx	5	Dyn.	264	565	8,78	94,9%	38,8%
nginx	25	Dyn.	266	2710	8,81	95,8%	38,7%
nginx	50	Dyn.	263	5187	8,73	95,6%	38,8%

Taulukko 3: nginx-sovelluksen rasitustestin tulokset

Nginx selvisi staattisten sivujen tarjoilusta parhaiten kolmesta testatusta palvelinsovelluksesta. Se ehti hoitaa 30 sekunnin aikana yli 200 000 sivupyynnön käsittelyajan pysyessä alle sekunnin sadasosan. Se käytti kyseisessä tilanteessa myös vähiten suoritinaikaa ja keskusmuistia.

Dynaamisia sivuja tarjoillessa nginx suoriutui hieman Apache2:sta paremmin, mutta tarjoiltujen sivujen määrä jäi silti merkittävästi pienemmäksi kuin Varnish-välityspalvelinta käytettäessä. Keskusmuistin käyttö nousi raskaimmassa testitilanteessa noin 39%:iin, mikä on tuntuvasti vähemmän kuin Apache2:n 76,2%.

RPI2:n suoritin vaikuttaa olevan nginxinkin tapauksessa suurin yksittäinen este. Mikäli RPI2 halutaan HTTP-palvelinkäyttöön, saattaisi nginxin ja varnishin yhdistelmä olla mahdollinen vaihtoehto, jolloin nginx tarjoilisi staattisen sisällön ja varnish dynaamisen.

Nginxin korkeat tulokset staattisen sivujen näyttämässä varmistettuun tietomäärältään suuremmalla sivulla, joka koostui kappaleesta 5.5.4 kuvastusta yksinkertaisesta HTML-sivusta ja sille lisätystä 653 kilotavun kokoisesta PNG-kuvasta. 50 yhtäaikaisen yhteyden testissä nginx ehti näyttää sivuja 223457 kertaa, ja kun kuva poistettiin jälleen, näyttöjä oli 226675 kappaletta. Näistä hajonnoista pääteltiin, ettei moninkertaistettu siirrettävän datan määrä vaikuttanut merkittävästi tuloksiin.

## 7 Pohdinta

Tässä luvussa tarkastellaan saatuja tuloksia, tutkimuksen luotettavuutta ja uskottavuutta sekä opinnäyteprosessia ja omaa oppimista.

### 7.1 Tulokset

Testattaviksi valitut palvelinsovellukset asentuivat Raspbian-käyttöjärjestelmässä paketinhallinnasta ilman virheilmoituksia. Merkillepantavaa oli, että useamman ohjelman oletusasetukset eivät toimineet ongelmitta (ISC DHCP server ja varnish). Osasy tähän lienee Debianin (ja siten Raspbianin) siirtyminen systemd-nimiseen ohjelmien ja palvelinten käynnistysprosessiin; kaikkien paketinhallinnasta saatavien ohjelmien dokumentaatio ei ole vielä ajan tasalla, eikä siirtymästä johtuvia bugeja ole toistaiseksi ehditty välttämättä korjata.

Kun valittujen sovellusten asetukset oli määritetty oikein, toimivat ne testiympäristössä toivotulla tavalla: Lähiverkon koneet saivat osoiteasetuksensa DHCP-palvelimelta ja nimihaut toimivat valituilla nimipalvelimilla. HTTP-palvelimet ja MySQL-palvelin toimivat ongelmitta ja RPI2 selvisi staattisten sivujen tarjoilemisesta hyvin.

Raspbianin sovellusvalikoima oli palvelinkäytön kannalta hyvä; laajalti käytettyjen ISC:n DHCP-palvelimen ja BIND9:n lisäksi saatavilla oli myös kevyempiä, pienempään käyttöön suunniteltuja vaihtoehtoja.

Apache2 ja nginx -palvelinsovellukset eivät pystyneet laitteen suorituskyvyn takia tarjoilemaan nykyaikaisia verkkosivuja kuin muutamia satoja minuutissa, tällöinkin käsittelystä johtuvan viiveen noustessa useisiin sekunteihin. Varnish-välityspalvelin paransi tosin dynaamisten sivujen tarjoilua dramaattisesti: tarjoiltujen sivujen määrä nousi kahdesta sadasta kahteenkymmeneen tuhanteen. Tämä parantaa RPI2:n soveltuvuutta HTTP-palvelimeksi, mutta tiedon tallennus on silti laitteen microSD-ratkaisun takia ongelmallinen. Palvelimen sisällön eheyden ja saatavuuden takaaminen tuotantoympäristössä olisi hankala toteuttaa.

Paremmilla liitännöillä varustettu Bananapi-tietokone saattaisi ratkaista osan tiedon tallennusongelmasta, mutta se ei ole yhtä tunnettu laite, jolloin tuen saaminen voi nousta omaksi ongelmakseen.

## 7.2 Tutkimuksen luotettavuus ja uskottavuus

Palvelin on laaja käsite ja tässä työssä käsitellään yhden koneen ja suhteellisesti vähän käytetyn käyttöjärjestelmän sovellusvalikoimaa. Tutkitut sovellukset sinänsä ovat yleisesti käytettyjä, mutta niiden toimintaa ei tässä työssä arvioida syvällisesti. Toisaalta niiden pitäisi olla toimintalogiikaltaan riippumattomia niitä suorittavasta tietokoneesta.

Tutkimustulosten luotettavuuden kannalta käytetty testiympäristö on toimiva. Vaikka työssä käytetty lähiverkko on konemäärältään pieni, voitiin sen avulla todeta palvelinsovellusten toimivan. Tutkimuksessa ei toisaalta oteta kantaa skaalautuvuuteen: RPI2 saattaa olla riittämätön ison yritysverkon DHCP- tai nimipalveluun.

Testiympäristössä oli Windows 7 -tietokone ja Linux Mint -kannettava, joilla saatiin RPI2:n asiakkaisiksi keskenään erilaisia tietokoneita. Mahdollinen ongelma on ympäristössä käytetty aktiivilaite. Vaikka sen suorittimen käyttöastetta pyrittiin tarkkailemaan, se ei ole välttämättä riittävä mittari mittausvirheiden karsimiseen.

Ab-testausohjelmalla ei välttämättä saatu selville kaikkia RPI2:n suorituskyvyn pullonkauloja, vaikka sillä saatiinkin vertailukelpoisia tuloksia kolmen eri HTTP-palvelinkonfiguraation välillä. Lisäksi sillä saaduissa tuloksissa oli Varnish-välityspalvelimen tarjoilemien staattisten sivujen kohdalla niin paljon hajontaa identtisissä testeissä, ettei niitä voida pitää luotettavina.

Testauksessa käytetty staattinen sivu oli hyvin pienikokoinen. Tämän vaikutusta tuloksiin arvioitiin lisäämällä nginx-palvelinsovelluksen kohdalla sivulle lisää dataa kuvatiedoston muodossa (ks. luku 6.3); vaikuttaa siltä, ettei pieni siirrettävän datan määrä vääristänyt tuloksia RPI2 ja Raspbian-yhdistelmän kohdalla.

## 7.3 Johtopäätökset ja ehdotukset

Vaikka sovellusten saatavuus on hyvä ja ne toimivat käytännössä testien perusteella, on RPI2 mahdollisena palvelinkoneena silti epävarma vaihtoehto. Muistikorttien luotettavuus ei tee siitä houkuttelevaa valintaa tiedostojen saatavuuden ja säilymisen kannalta.

HTTP-palvelimeksi RPI2 ei tutkimuksen perusteella sovellu järin hyvin. Raskaiden dynaamisten verkkosivujen tarjoilemisessa se jää varsinaisten palvelinkoneiden varjoon suorituskyvynsä takia, vaikka välityspalvelimella voidaankin tilannetta parantaa. Staattisia sivuja se pystyy näyttämään suuriakin määriä, mutta staattiset sivut eivät ole monelle

toimijalle houkutteleva sivutyyppe. RPI2:n hankinnan kokonaiskustannukset jäivät noin sadan euron hintaluokkaan, mutta palvelun esimerkiksi yrityksen verkkosivuja varten voi myös vuokrata internetistä edullisesti (Aakala 2009, 302).

DHCP- ja nimipalvelimenä se on järkevämpi vaihtoehto, mutta DHCP-palvelu löytyy monista aktiivilaitteista ja nimipalvelulle ei tarvita välttämättä erillistä tietokonetta. Keskusmuistin määrän lisääminen ei ole myöskään toteutettavissa järkevästi, mikä haittaa palvelinkäyttöä, sillä keskusmuistin määrä on palvelimen toimintakyvyn kannalta olennainen muuttuja (Aakala 2009, 129).

Resurssien käyttöä tarkkaillessa kävi ilmi, että laitteen suorittimen käyttö nousee nopeasti korkeaksi. Vaikka HTTP-palvelin ei olekaan sille hyvin soveltuva käyttötarkoitus, on kevyempiäkin palvelinkäyttötarkoituksia. Tällaisiin RPI2 saattaa ollakin hyvä vaihtoehto.

#### **7.4 Opinnäytetyöprosessi**

Opinnäytetyötä tehdessä oli mielenkiintoista huomata, kuinka suunnitelluissa aikatauluissa oli vaikea pysyä. Tämä johtui osittain optimistisista aika-arvioista eri työn osa-alueille. Yksittäisistä asioista mainittakoon ab-testien tulosten analysointi ja purkaminen järkevästi vertailtavaan muotoon ja varsinainen tekstinkäsittely. Odotettua enemmän aikaa kului myös käyttökelpoisten lähteiden etsimisessä. Ajan tasalla olevia kirjallähteitä ei löytynyt RPI2 tai Raspianiin liittyen.

Opinnäyteprosessin suunniteltu perusrakenne (aihe-ehdotus, projektisuunnitelma, teoriapohja, empiirinen osuus) helpotti työn tekemistä. Työn sisältö tosin eli jonkin verran; projektisuunnitelmassa oli alun perin suunniteltu, että RPI2:n kustannuksia arvioitaisiin verrattuna muihin oletettavasti edullisiin ratkaisuihin, kuten palvelintilan vuokraamiseen, josta projektin aikana kuitenkin luovuttiin, sillä tutkimuksessa keskityttiin enemmän RPI2:n toimintaan käytännössä.

Projektisuunnitelman riskikartoituksessa arvioitiin merkittävimiksi riskeiksi projektipäällikön (opinnäytetyön tekijä) sairastuminen ja laitteiston hajoaminen. Kumpikin riski toteutui, tosin suunnitelmassa ennustettu microSD-kortin hajoaminen ei tapahtunut, vaan itse RPI2 vaurioitui huolimattoman kotelosta poistamisen seurauksena. Suunnitelmassa oli varattu riskien takia 5 viikkoa aikaa projektin ennustetun loppumispäivän ja viimeisen arviointikokouspäivän väliin, mikä osoittautui hyväksi asiaksi, sillä aikataulu oli loppujen lopuksi melko tiukka.

Haastavimmiksi seikoiksi opinnäytetyöprosessissa osoittautuivat aiheen keksiminen, koska varsinaista toimeksiantajaa ei löytynyt. Aiheen löydyttyä sen purkaminen toteuttamiskelpoiseksi opinnäytetyöksi ja kokonaisuuden hahmottaminen veivät myös odotettua enemmän aikaa.

Ohjauskokoukset ja seminaari auttoivat huomattavan paljon opinnäytetyön teossa. Välissä saatu palaute ja vertaisarviointi vaikuttavat sinällään työn sisältöön, mutta olivat käytännössä tärkeitä myös mielenrauhan kannalta, sillä niiden avulla sai käsityksen työn senhetkisestä tilasta. Stressinhallinnan huomioon ottaminen oli ylipäätään tärkeää työn valmistumisen kannalta.

## **7.5 Oma oppiminen**

Opinnäytetyön tekeminen oli opettava kokemus projektinhallinnan harjoituksena. Vaikka ammattikorkeakouluopintojen aikana onkin tehty laajempia kirjallisia selvityksiä, oli opinnäytetyö niistä haastavin niin haasteiden kuin työmäärän kannalta sinänsä.

Aikataulun suunnitteleminen ja noudattaminen osoittautuivat tärkeiksi asioiksi projektinhallinnan kannalta; ongelmien purkaminen pienempiin paloihin ja kokonaiskuvan hahmottaminen sitä kautta oli niin ajankäytön kuin työn sisällönkin kannalta hyvä toimintatapa. Kun selvitettävät asiat muotoili erillisiksi kysymyksiksi ja etsi vastauksia lähteiden ja empirian kautta, työn rakenne alkoi selkiytyä. Jos opinnäytetyö pitäisi tehdä nyt uudelleen, tulisi aikataulutukseen kiinnitettyä enemmän huomiota.

Valitsin työn aiheeksi Raspberry Pi 2 -tietokoneen ja Raspbianin, koska olen kiinnostunut Linuxista yleisesti. Haaga-Helia ammattikorkeakoulun tarjoamat syventävät opinnot ovat tarjonneet hyvän pohjan itsenäiselle ja mahdolliselle jatko-opiskelulle, ja opinnäytetyötä tehdessä opittuja asioita pääsi soveltamaan käytännössä. Verkon tietokoneiden etähallinta ja terminaalin käyttö alkoivat sujua työn aikana luontevasti. Erilaisen DHCP- ja nimipalvelinten käyttöönotto ja konfigurointi testiympäristössä eivät sujuneet kauttaaltaan ennako-odotusten mukaisesti. Niiden asennus oli ongelmaton, mutta asetusten määrittelyssä oli ongelmia, joiden selvittämiseen meni aikaa. Olikin palkitsevaa saada ne ennen pitkään toimintakuntoon. Ottaen huomioon, että ISC dhcp server ja BIND ovat varsin käytettyjä palvelinsovelluksia, arvelen opinnäytetyön aikana tehdyn ongelmanratkaisun olevan hyvää kokemusta.

Testiympäristön suunnittelu tarkoituksenmukaiseksi ja sen toteutus opinnäytetyössä vastasivat mittakaavaltaan syventäviä järjestelmäasiantuntijan polun kursseja. Sen sijaan rasiustestien toteuttaminen oli yllättävän työläs ja lopulta palkitseva tehtävä.

Tutkimuksessa käytettyä ab-rasitustestausohjelmaa voi käyttää useilla parametreilla, minkä lisäksi tarkasteltavien muuttujien valikoiminen tuloksista ja johtopäätösten tekeminen niiden perusteella oli haastavaa. Koska suorituskyvyn optimoinnilla saavutetaan monesti rahallisia säästöjä ja se edellyttää nykytilan arviointia, oli rasitustestauksen suunnitteluun ja toteutukseen perehtyminen työelämää ajatellen todennäköisesti hyvää harjoitusta.

## Lähteet

Aakala, K. & Asp, R. 2009. Järjestelmätuen peruskirja. WSOYproOy. Jyväskylä.

Apache. Apache HTTP server benchmarking tool. Luettavissa:

<https://httpd.apache.org/docs/2.2/programs/ab.html>. Luettu: 18.10.2015.

ApacheTutor 2009. Running a Reverse Proxy in Apache. Luettavissa:

<http://www.apachetutor.org/admin/reverseproxies>. Luettu 16.10.2015.

Armstrong, A. 2014. Debian And The Systemd Storm - Ready To Reconsider?.

Luettavissa: <http://www.i-programmer.info/news/126-os/7882-debian-and-the-systemd-storm-ready-to-reconsider.html>. Luettu: 14.10.2015.

Bananapi 2014. What is Banana Pi?. Luettavissa: <http://www.bananapi.org/p/product.html>.

Luettu: 14.10.2015.

Crump, G. 2011. Demystifying SSD Wear Leveling. Luettavissa:

<http://www.networkcomputing.com/storage/demystifying-ssd-wear-leveling/a/d-id/1097528?>.

Debian. Package: dnstools (1:9.9.5.dfsg-9+deb8u3) [security]. Luettavissa:

<https://packages.debian.org/jessie/dnstools>. Luettu 19.10.2015.

Debian 2015. Debian 8 Jessie released. Luettavissa:

<https://www.debian.org/News/2015/20150426>. Luettu 14.10.2015.

Ellingwood, J. 2014a. A Comparison of DNS Server Types: How To Choose the Right

DNS Configuration. Luettavissa: <https://www.digitalocean.com/community/tutorials/a-comparison-of-dns-server-types-how-to-choose-the-right-dns-configuration>. Luettu

16.10.2015.

Ellingwood, J. 2014b. How To Configure Bind as a Caching or Forwarding DNS Server on

Ubuntu 14.04. Luettavissa: <https://www.digitalocean.com/community/tutorials/how-to-configure-bind-as-a-caching-or-forwarding-dns-server-on-ubuntu-14-04>. Luettu 5.11.2015.

Gonyea, C. 2010. DNS: Why It's Important & How It Works. Luettavissa:

<http://dyn.com/blog/dns-why-its-important-how-it-works>. Luettu 16.10.2015.

Hintaseuranta. 2015. Micro Secure Digital High Capacity (microSDHC) 8GB, muistikortti. Luettavissa: <http://hintaseuranta.fi/tuote/micro-secure-digital-high-capacity-microsdhc-8gb-muistikortti/75714>. Luettu 14.10.2015.

Instructables. 2015. Remote SSH access to Raspberry Pi 2. Luettavissa: <http://www.instructables.com/id/Remote-SSH-access-to-Raspberry-Pi-2>. Luettu 18.10.2015.

ISC 2015a. DHCP Subscription. Luettavissa: <https://www.isc.org/dhcp-subscription>. Luettu 16.10.2015.

ISC 2015b. BIND. Luettavissa: <https://www.isc.org/downloads/bind>. Luettu: 16.10.2015.

Jackson, J. 2008. SDHC Cards vs Hard Drive vs SSD. Luettavissa: <http://www.notebookreview.com/news/sdhc-cards-vs-hard-drive-vs-ssd>. Luettu: 14.10.2015.

James, H. 2015. Raspberry Pi 2 Overclock. Luettavissa: <http://haydenjames.io/raspberry-pi-2-overclock>. Luettu 16.10.2015.

Koski, R. 2010. Linux - Tehokas hallinta. Readme.fi. Helsinki.

Linuxconfig. What is DHCP and how to configure DHCP server in Linux. Luettavissa: <http://linuxconfig.org/what-is-dhcp-and-how-to-configure-dhcp-server-in-linux>. Luettu 16.10.2015.

Modmypi. 2013. Tutorial - How to give your Raspberry Pi a Static IP Address. Luettavissa: <https://www.modmypi.com/blog/tutorial-how-to-give-your-raspberry-pi-a-static-ip-address>. Luettu 18.10.2015.

Nicholson, J. 2014. Dynamic website vs Static website. Luettavissa: <http://www.inmotionhosting.com/support/website/slow-websites/dynamic-website-vs-static-website>. Luettu 16.10.2015.

Noveldevices. Setting up a Raspberry Pi as a DHCP server. Luettavissa: <http://www.noveldevices.co.uk/rp-dhcp-server>. Luettu 5.11.2015.

Raspberrypi 2013. Debian Security Updates. Luettavissa: <https://www.raspberrypi.org/forums/viewtopic.php?f=66&t=54442>. Luettu 14.10.2015.



Raspberrypi a. What is a Raspberry Pi?. Luettavissa:  
<https://www.raspberrypi.org/help/what-is-a-raspberry-pi>. Luettu: 13.10.2015.

Raspberrypi b. Raspberry Pi FAQs – Frequently Asked Questions. Luettavissa:  
<https://www.raspberrypi.org/help/faqs>. Luettu: 13.10.2015.

Raspberrypi c. Raspberry Pi 2 Model B. Luettavissa:  
<https://www.raspberrypi.org/products/raspberry-pi-2-model-b>. Luettu 13.10.2015.

Raspberrypi d. GPIO: Raspberry Pi Models A and B. Luettavissa:  
<https://www.raspberrypi.org/documentation/usage/gpio>. Luettu 13.10.2015.

Raspberrypi e. Installing operating system images. Luettavissa:  
<https://www.raspberrypi.org/documentation/installation/installing-images>. Luettu  
18.10.2015.

Raspberrypi 2015. Jessie Is Here. Luettavissa: <https://www.raspberrypi.org/blog/raspbian-jessie-is-here>. Luettu 14.10.2015.

Raspbian a. FrontPage – Raspbian. Luettavissa: <https://www.raspbian.org>. Luettu  
13.10.2015.

Raspbian b. RaspbianFAQ. Luettavissa: <https://www.raspbian.org/RaspbianFAQ>. Luettu  
13.10.2015.

RasPi. 2014. How Much Less Power does the Raspberry Pi B+ use than the old model B?. Luettavissa: <http://raspi.tv/2014/how-much-less-power-does-the-raspberry-pi-b-use-than-the-old-model-b> Luettu: 14.10.2015.

Stackoverflow. 2013. nginx showing blank PHP pages. Luettavissa:  
<https://stackoverflow.com/questions/15423500/nginx-showing-blank-php-pages>. Luettu:  
5.11.2015.

TechTerms. 2015a. Static Website. Luettavissa:  
<http://techterms.com/definition/staticwebsite>. Luettu 16.10.2015.

TechTerms. 2015b. Dynamic Website. Luettavissa:  
<http://techterms.com/definition/dynamicwebsite>. Luettu 16.10.2015.

TheKelleys. Dnsmasq - network services for small networks. Luettavissa: <http://www.thekelleys.org.uk/dnsmasq/doc.html>. Luettu 16.10.2015.

Thompson, R. & Thompson, B. 2003. PC Hardware in a Nutshell. O'Reilly. Sebastopol.

Ubuntu 2015. Ubuntu and Debian. Luettavissa: <http://www.ubuntu.com/about/about-ubuntu/ubuntu-and-debian>. Luettu: 14.10.2015.

Peters, M. 2015. Average Power Use Per Server. Luettavissa: <http://www.vertatique.com/average-power-use-server>. Luettu: 14.10.2015.

Vance, A. 2014. ARM Designs One of the World's Most-Used Products. So Where's the Money?. Luettavissa: <http://www.bloomberg.com/bw/articles/2014-02-04/arm-chips-are-the-most-used-consumer-product-dot-where-s-the-money>. Luettu: 14.10.2015.

WhatIs 2015. Web server. Luettavissa: <http://whatis.techtarget.com/definition/Web-server>. Luettu 16.10.2015.

Winkler, C. 2012. Ultra-Performant Dynamic Websites with Varnish. Luettavissa: <http://blog.mgm-tp.com/2012/01/varnish-web-cache>. Luettu 16.10.2015.

Yiqingsim. 2014. Setting up dnsmasq as a DNS/DHCP server on a Raspberry Pi. Luettavissa: <http://yiqingsim.net/post/103165692292/setting-up-dnsmasq-as-a-dnsdhcp-server-on-a>. Luettu: 5.11.2015.

## Liitteet

### Liite 1. Esimerkki Apache Benchmark -ohjelman tuloksista

```
$ ab -t 30 -n 1000000 -c 50 -k -s 10000 http://192.168.1.33/
This is ApacheBench, Version 2.3 <$Revision: 1604373 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/
```

Benchmarking 192.168.1.33 (be patient)

Completed 100000 requests

Completed 200000 requests

Finished 223457 requests

```
Server Software:      nginx/1.6.2
Server Hostname:     192.168.1.33
Server Port:         80
```

```
Document Path:       /
Document Length:     129 bytes
```

```
Concurrency Level:   50
Time taken for tests: 30.000 seconds
Complete requests:   223457
Failed requests:     0
Keep-Alive requests: 221245
Total transferred:   81550745 bytes
HTML transferred:   28825953 bytes
Requests per second: 7448.55 [#/sec] (mean)
Time per request:    6.713 [ms] (mean)
Time per request:    0.134 [ms] (mean, across all concurrent requests)
Transfer rate:       2654.64 [Kbytes/sec] received
```

#### Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.1	0	6
Processing:	1	7 9.3	5	344
Waiting:	0	7 9.3	5	344
Total:	1	7 9.3	5	349

Percentage of the requests served within a certain time (ms)

50% 5

66%	7
75%	8
80%	8
90%	10
95%	12
98%	22
99%	44
100%	349 (longest request)