Ville Lehtola

Kustannustehokas pilvipalveluratkaisu pk-yrityksen käyttöön

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikka

Insinöörityö

22.05.2016

Metropolia

Tekijä(t) Otsikko Sivumäärä Aika	Ville Lehtola Kustannustehokas pilvipalveluratkaisu pk-yrityksen käyttöön 48 sivua + 1 liitettä 22.5.2016
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Tietoverkot
Ohjaaja(t)	Osaamisaluepäällikkö Janne Salonen

Pilvipalvelut eivät ole enää uusi käsite, niiden suosio on kasvanut voimakkaasti viimeisten vuosien aikana. Tämä tutkielma käsittelee pilvipalveluiden eri toteutusmuotoja. Tutkielmassa perehdytään erityyppisten pilvipalveluiden toimintaan sekä selvitetään niiden avulla saatavia hyötyjä ja haittoja.

Insinöörityön päätavoitteena oli toteuttaa Infrastruktuuri palveluna -mallinen pilvipalveluratkaisu mahdollisimman kustannustehokkaasti. Palvelu rakennettiin suljettuun ympäristöön asiakkaan verkossa. Näin toimittiin, jotta ulkopuoliset tahot eivät päässeet näkemään esimerkiksi uudistettavaa verkkosivustoa tai muita palvelun osioita työn tekemisen aikana.

Työssä esiteltiin tekniseltä toteutukseltaan erilaisia pilvipalvelumalleja ja otettiin kantaa eri mallien tietoturvaan. Nämä palvelumallit ovat nimeltään: infrastruktuuri palveluna (IaaS, Infrastructure as a Service), alusta palveluna (PaaS, Platform as a Service) ja ohjelmisto palveluna (SaaS, Software as a Service). Palvelumallien avulla jaettiin erilaiset pilvipalvelut lukijalle helposti ymmärrettäviin osioihin.

Eri mallit voidaan jakaa myös toimitusmallien mukaisiin luokkiin. Mallit jaettiin luokkiin niiden tietoturvaominaisuuksien perusteella. Työssä esitellyt toimitusmallit ovat julkinen pilvi (Public cloud), yksityinen pilvi (Private cloud) ja hybridi pilvi (Hybrid cloud). Tekniseen toteutukseen valittiin yksityisen pilven toimitusmalli.

Osana työtä esiteltiin fyysistä laitteistoa ja vertailtiin eri ARM-pohjaisia minitietokoneita. Laitteistovertailun jälkeen päädyttiin hyödyntämään Raspberry Pi -minitietokonetta palvelun alustana.

Käytännön toteutuksessa keskitytään Infrastruktuuri palveluna -mallisen pilvipalvelun toteuttamiseen ja toteutuksen vaatimien työvaiheiden suorittamiseen. Työhön sisällytettiin verkkosivujen uudistaminen ja hallinta-alustan vaihtaminen, yksityinen pilvitallennusratkaisu ja sähköpostipalvelut. Lopuksi on pohdittu työn onnistumista, ongelmatilanteita ja mahdollisia kehitystoimenpiteitä.

Avainsanat	Pilvipalvelut, pilvilaskenta, Raspberry Pi, Word Press, Owncloud

Author(s) Title Number of Pages Date	Ville Lehtola Cost-effective cloud service solution for small business 48 pages + 1 appendices 22 May 2016
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Data Networks
Instructor(s)	Janne Salonen, Head of Department

Cloud computing isn't a new concept and its popularity has increased significantly over the last few years. This study examines most common cloud computing models and their technical aspects, functions, benefits and disadvantages.

Goal of this thesis was to build fully operational cloud service for a customer, using Infrastructure as a Service cloud computing model. Service was built into private cloud, so that third parties couldn't access the information during the development phase.

Thesis presents three most common cloud service models focusing on their security aspects. These services are: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Service models are used to categorize different services into easily understandable format for the reader.

Different models can be categorized by their technical implementation. Categories covered in this thesis are: Public cloud, Private cloud, Hybrid cloud. In this thesis, private cloud model was used for the implementation, for its security properties.

One part of the thesis was to compare different hardware and especially ARM based computers that could be used for the implementation. Based on the comparison Raspberry Pi was selected to be used as a platform.

Study focuses on setting up an IaaS cloud service and describing required steps in implementation process, including customer's web page layout renewal and changing the management platform, private cloud solution and email services.

Summary includes evaluation of the implementation process and presents different possibilities for further development of the service.

Keywords	Cloud services, Cloud computing, Rasperry Pi, Word Press, Owncloud

Sisällys

Lyhenteet

RPi	Raspberry Pi. ARM-pohjainen minitietokone.
VMM	Virtual Machine Manager. Virtualisointialusta.
RAM	Random access memory. Keskusmuisti.
AWS	Amazon Web Services. Amazonin tarjoama pilvipalvelualusta.
SSH	Secure Shell. Tietoliikenteen salaukseen käytettävä protokolla.
SSL	Secure Socket Layer. Salausprotokolla, jolla suojataan tietoliikenne internetissä.
TLS	Transport Layer Security. SSL:n seuraava versio.
PHP	Hypertext Preprocessor. Ohjelmointikieli, jota käytetään dynaamisten verkkosivujen luonnissa.
IP	Internet Protocol. Protokolla, joka huolehtii pakettien toimittamisesta oikeaan paikkaan.
NOOBS	New Out Of Box Software. Käyttöjärjestelmäasennusta helpottava työkalu.
PuTTY	Telnet- ja SSH pääte-emulaattori.
SQL	Structured Query Language. Relaatiotietokanta.
WordPress	Verkkosivustojen julkaisualusta.
CSS	Cascading Style Sheets. Verkkosivun tyyliohjeiden laji.
DNS	Domain Name System. Internetin nimipalvelujärjestelmä.

1 Johdanto

Pilvipalvelut eivät ole enää uusi käsite, niiden suosio onkin kasvanut voimakkaasti viimeisten vuosien aikana. Monelle on kuitenkin epäselvää, mitä pilvipalvelut tarkoittavat ja mihin niitä voidaan käyttää. Pilvipalveluilla ei ole vielä vakiintunutta, yksikäsitteistä määritelmää. Pilvipalveluksi voidaan ajatella kaikkea verkkopohjaista tiedonhallintaa, jossa ohjelmistot, laitteistot ja käyttäjien tiedostot sijaitsevat verkossa eivätkä käyttäjän tietokoneella.

Pilvipalvelut ovat tuoneet mukanaan uusia ajattelutapoja ja mahdollistaneet tehokkaamman työskentelymallin. Ne mahdollistavat työn tekemisen mistä vain ja milloin vain. Tämä on suoranaisesti vaikuttanut yritysten toimintatapoihin. Esimerkiksi työn tekeminen kotona on yleistynyt huomattavasti pilvipalveluiden ansiosta. Pilvipalveluita tarjoavat useat kansainväliset yritykset mm. Microsoft, Amazon, Salesforce, Google ja Oracle.

Insinöörityön tekijä työllisti itsensä ehdottamalla pilvitoteutusta AC-Tekniikka ky:lle kustannustehokkaamman IT-infran saavuttamiseksi. AC-tekniikka ky myy, asentaa ja huoltaa lämpöpumpputekniikkaan perustuvia energiatehokkaita lämmitys- ja jäähdytysjärjestelmiä kuluttajille ja yritysasiakkaille.

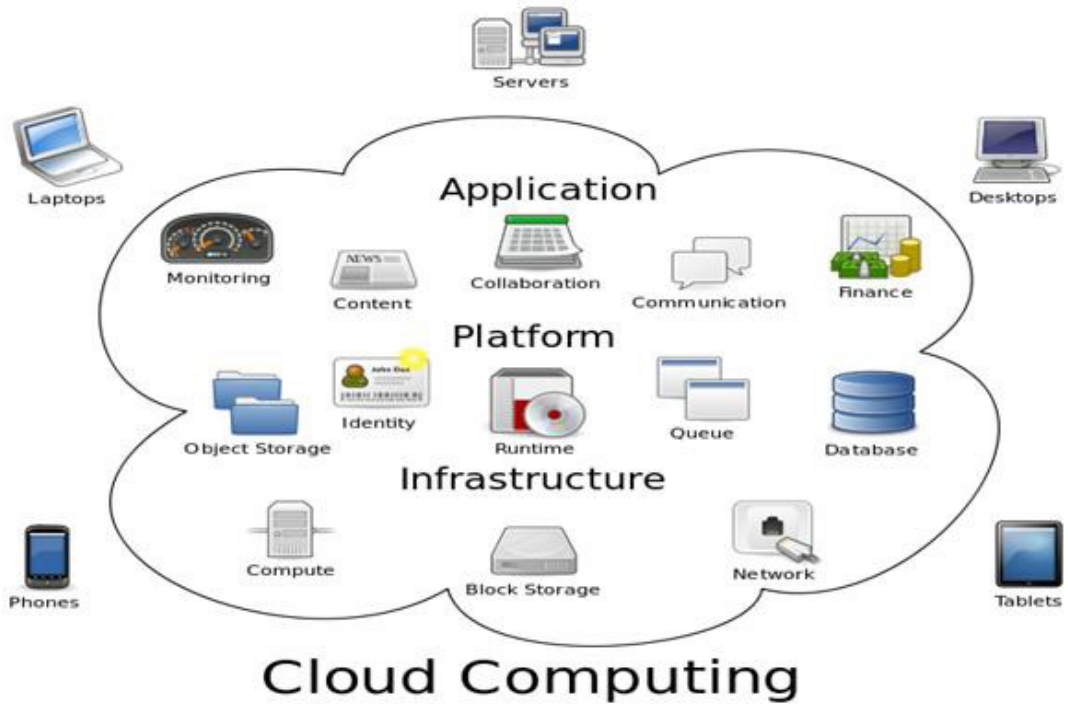
Tässä insinöörityössä toteutettiin pk-yrityksen tarpeisiin sopiva IaaS-pilvipalveluympäristö. Palvelu rajattiin niin, ettei ulkopuolisilla ollut mahdollisuutta nähdä palvelun sisältöä rakennus- ja testausvaiheessa.

Työtä suunniteltaessa keskityttiin alkuinvestoinnin ja käyttöönotetun ympäristön kuluihin silmälläpitäen käyttövarmuutta ja helppoutta. Työhön sisällytettiin verkkosivujen uudistaminen ja hallinta-alustan muuttaminen helppokäyttöisempään vaihtoehtoon, yksityinen pilvitallennusratkaisu ja sähköpostipalvelut.

2 Pilvipalvelut

Pilvipalveluilla (cloud computing) tarkoitetaan internetin kautta käytettäviä sovelluksia, laskentakapasiteettia tai muita palveluita. Pilvi on erilaisten palveluiden toteutustekniikka, jossa esimerkiksi ohjelmistoja käytetään Internetin yli palvelun tarjoajan konesalista, eikä perinteisesti loppukäyttäjän tietokoneelta.

Pilvipalvelut ovat saaneet nimensä arkkitehtuurikuvien pilvisymbolista, jolla kuvattiin isoja tai hankalasti esitettäviä verkkoja, kuten WAN (Wide area network). Koska pilvipalvelut tuotetaan datakeskuksissa ja niitä käytetään Internetin yli, piirretään ne kaavioihin pilvisymbolin sisään.



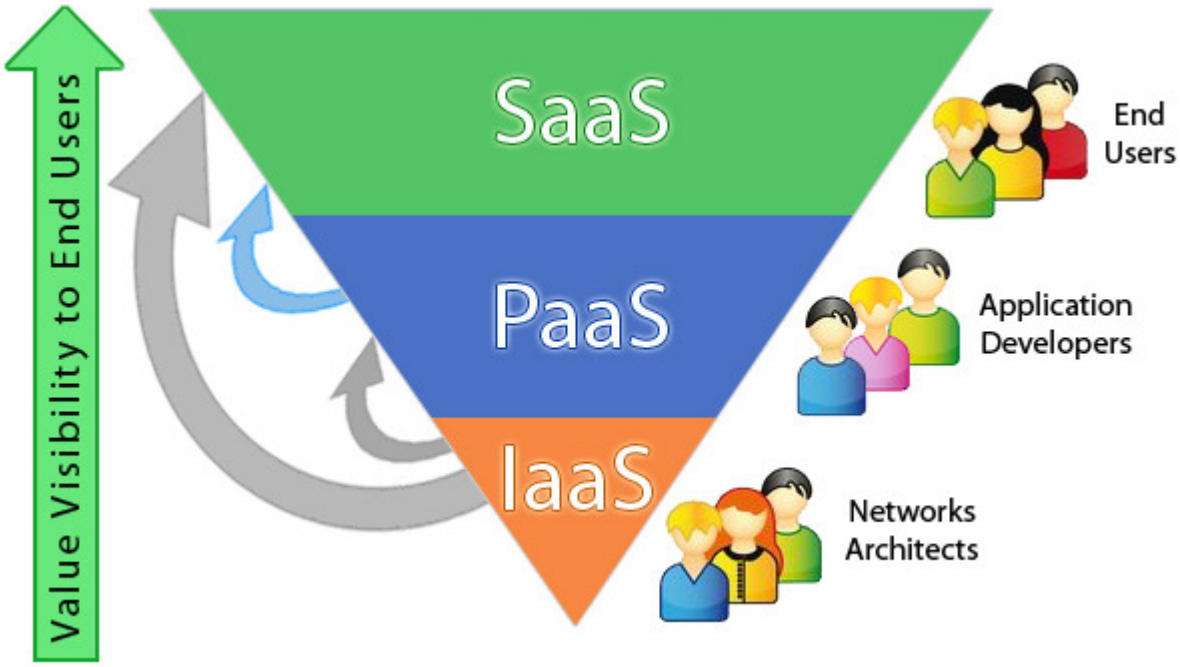
Kuva 1, Pilvipalveluita yhdistettynä liityntälaitteisiin.

Pilvipalvelut yleistyivät 2010-luvulla, niiden suosio on kasvanut räjähdysmäisesti niin yritysten kuin kotikäyttäjien keskuudessa. Ne muokkaavat IT-palvelualaa uuteen suuntaan mahdollistaen uusia bisnesmalleja ja tapoja IT-infran toteuttamiseen. Pilvilaskennan suosiota on nostanut sen käyttöönoton helppous, kustannustehokkuus sekä skaalautuvuus erikokoisten yritysten ja organisaatioiden tarpeisiin. Kuluttajille suunnatuissa

pilvipalveluissa yleensä peruspaketti on ilmainen, mutta lisäkapasiteetti ja ominaisuudet maksavat.

3 Pilvipalveluiden palvelumallit

Pilvipalvelut luokitellaan muutamaan palvelumalliin teknisen toteutustavan perusteella. Luokittelu kertoo, millaisia tietojenkäsittelytehtäviä pilvipalvelusta saadaan ja miten palveluun liitytään. Nämä palvelumallit ovat nimeltään: infrastuktuuri palveluna (Iaas, Infrastructure as a Service), alusta palveluna (Paas, Platform as a Service) ja ohjelmisto palveluna (SaaS, Software as a Service). Palvelumallit auttavat jakamaan erilaiset pilvipalvelut helpommin ymmärrettäviin osioihinsa.

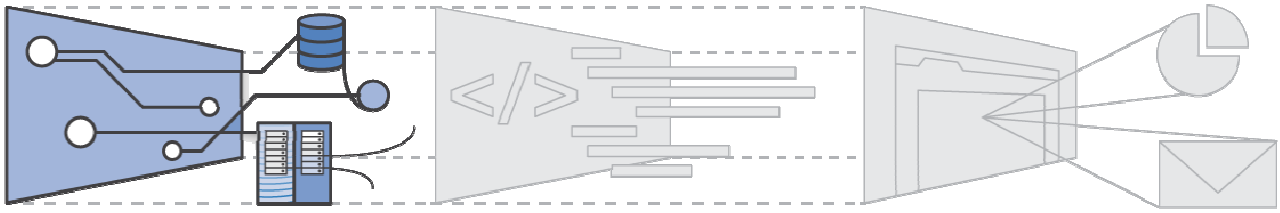


Kuva 2, Pilvipalvelumallit ja niiden yleisimmät käyttäjät.

3.1 Infrastruktuuri palveluna, IaaS

Infrastruktuuri palveluna (Infrastructure as a Service), tarkoittaa käytännössä palveluntarjoajalta vuokrattavaa rautaa, joka sijaitsee palveluntarjoajan konesaleissa. Palveluntarjoaja ylläpitää konesalia, josta se lohkoo etukäteen räätälöityjä ja hinnoiteltuja osioita asiakkaan käyttöön. Palveluntarjoaja vastaa fyysisistä laitteista, kuormituksen tasaajista, palvelun kahdentamisesta (jakamisesta useammalle fyysiselle laitteelle), skaalamisesta, turvallisuudesta, varmuuskopioinnista ja niin edelleen.

IaaS-pilven käyttö tapahtuu virtuaalialustojen VMM (Virtual Machine Manager) avulla, joiden resursseista asiakas päättää itse. Laskentatehot, keskusmuisti sekä tallennustila varataan virtuaalikoneelle, jota voidaan mukauttaa asiakkaan tarpeiden mukaan reaaliajassa ilman palvelukatkoksia. Esimerkkejä VMM-palveluista ovat: VMware ESXi, Hyper-V ja Xen.



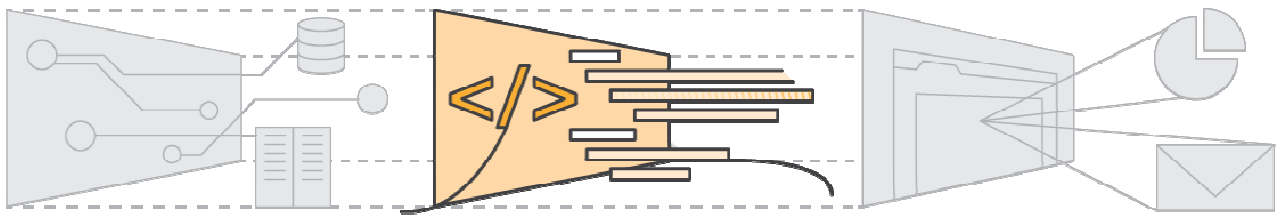
Kuva 3, IaaS-palvelumalli on lähimpänä perinteistä IT-infraa.

3.2 Alusta palveluna, PaaS

Alusta palveluna (Platform as a Service) tarkoittaa kysynnän mukaan skaalautuva ohjelmistokehitysalustaa pilvessä. PaaS-palveluntarjoajat tarjoavat työkalupakin ja standardit sovelluskehittämistä, jakelua ja maksamista varten.

Monien PaaS-palveluiden kohdalla, kuten Microsoft Azuren, Amazon AWS:n ja Google App Enginen kehittäjien ei itse tarvitse huolehtia ohjelmiston skaalautuvuudesta tai kehitetyn ohjelmiston käyttäjämäärien kasvusta, vaan alusta laajenee sovelluksen ja käyttäjämäärien mukaan automaattisesti.

Asiakas käyttää PaaS-palveluntarjoajan alustaa API-ohjelmointirajapinnan välityksellä ja tekee tai teettää palvelua hyödyntävät sovellukset. Asiakkaan käyttöliittymä on ohjelmistokehitysväline ja jonkinlainen hallintakonsoli, jota hallitaan asiakkaan omalta laitteelta. PaaS-alustaa käytetään yleensä selaimen avulla.



Kuva 4, PaaS-palvelumalli on tarkoitettu sovelluskehittäjille.

3.3 Ohjelmisto palveluna, SaaS

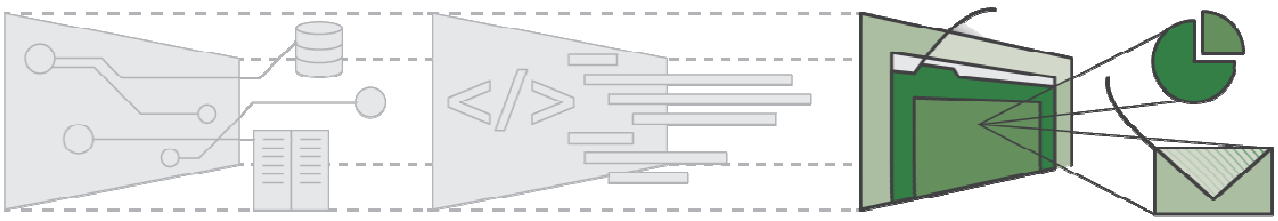
Ohjelmisto palveluna (Software as a Service) on ohjelmistojakelumalli, jossa myyjä tarjoaa sovelluksia usein selainkäyttöisinä, kuukausilaskutteisina ja välittömästi käyttöönotettavina. Saas-ohjelmistoa ei tarvitse asentaa paikallisesti koneelle, vaan sitä käytetään Internetin yli myyjän konesalista.

Pilvipalvelumalleista SaaS on tunnetuin. Saas-malli perustuu siihen, että ohjelmisto vuokrataan käyttöön ja maksetaan käytön tai käyttäjämäärien mukaan, minkä vuoksi SaaS:in käyttö on kustannustehokasta ja joustavaa. Vastuu tuotteen asennus-, ylläpito- ja huoltotoimista jää palveluntarjoajalle, loppukäyttäjä pystyy keskittymään ainoastaan sovelluksen käyttöön.

SaaS:in etuina ovat sen helppo käyttöönotto, hallinnointi, skaalautuvuus, automaattiset päivitykset ja palvelun toiminnan luotettavuus. SaaS-käyttäjän ei myöskään tarvitse huolehtia lisensoinnista, koska se hoidetaan palveluntarjoajan toimesta. SaaS ohjelmistoja voi käyttää melkein miltä tahansa laitteelta, kunhan laitteessa on internetyhteys.

Pilvisovellukset eroavat paikallisista sovelluksista niiden skaalautuvuuden mukaan. Skaalautuvuudella tarkoitetaan esimerkiksi tehtävien kloonaamista useille eri virtuaalilaitteille samanaikaisesti. Tällä tavoin vasteajat pienenevät, palvelu pystyy vastaamaan käyttäjän muuttuviin resurssitarpeisiin joustavammin.

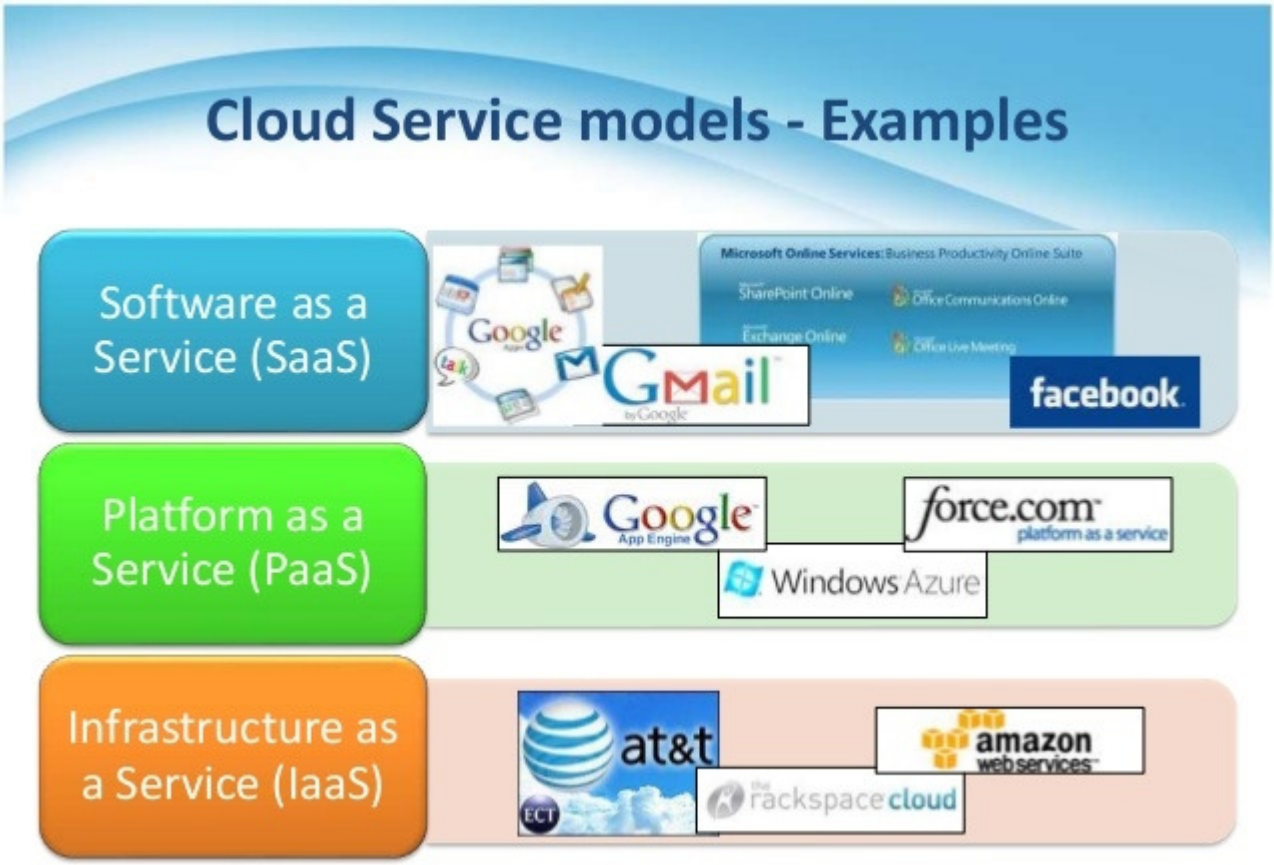
Yhdeltä palvelimelta voidaan palvella jopa kymmeniä tuhansia käyttäjiä samanaikaisesti, ohjaamalla laskentatehoa käyttäjien tarpeen mukaan. Käyttäjän näkökulmasta taustalla tehtävä laskentatyö ei näy lainkaan, vaan hän näkee ainoastaan yhden virtuaalipäätteen, jolla hän työskentelee.



Kuva 5, SaaS-palvelumallissa käyttäjälle näytetään ainoastaan virtuaalipääte.

SaaS on kustannustehokas vaihtoehto. Sillä vähennetään IT-kustannuksia ulkoistamalla laitteistojen ja ohjelmistojen ylläpitoa, tukipalveluita ja huoltoa palveluntarjoajille. Tämä taas tarkoittaa suoria säästöjä asiakkaalle laitteisto-, ohjelmisto- ja henkilöstökustannuksissa. Ohjelmistopäivitykset hoidetaan palveluntarjoajan puolelta, joten käyttäjillä on aina käytössään uusin ja tärkeänä etuna sama versio ohjelmistosta.

Haittapuolena SaaS-palvelumallissa on, että käyttäjän data tallennetaan pilveen. Tällöin loppukäyttäjällä ei ole tietoa, missä dataa maantieteellisesti säilytetään. Asiakkaan tiedot ovat jatkuvasti yhteydessä Internettiin, jolloin ne ovat alttiimpana mahdollisille tietomurroille ja kansainväliselle tietojenkalastelulle. [1.]



Kuva 6. Esimerkkejä palveluista.

4 Pilvipalveluiden toimitusmallit

Yleisiä pilvityyppejä on neljä kappaletta. Näistä kolme on enemmän käytössä ja yksi vähemmän tunnettu toimitusmalli. Nämä tyypit ovat: julkinen pilvi (Public cloud), yksi-

tyinen pilvi (Private cloud), hybridi pilvi (Hybrid cloud) ja yhteisöllinen pilvi (Community cloud). Malli määritellään sen tietoturva-arvojen perusteella. Julkisen pilven käyttö yrityksissä on puhtaasti kustannuskysymys. Arkaluontoista dataa käsitellessä käytetään yleensä yksityistä pilveä. Näiden kahden ratkaisun välissä toimii hybridi pilvi, joka on yhdistelmä molemmista toimitusmalleista. Yhteisöllinen pilvi on vähemmän käytetty malli, joka on myös yhdistelmä yksityisen ja julkisen pilven mallista, mutta sitä ylläpidetään useamman toimijan puolesta.

4.1 Julkinen pilvi

Julkinen pilvimalli on tunnetuin toimitusmalli. Siinä palveluntarjoaja tarjoaa virtuaalisia resursseja, kuten sovelluksia tai tallennustilaa käyttäjille julkisen internetin välityksellä. Nämä palvelut voivat olla joko ilmaisia tai maksaa käytön ja laajuuden mukaan. Julkisen pilven hyöty saadaan sen helppoudesta, kustannustehokkuudesta ja käyttövarmuudesta.

Kuva 6, Julkiseen pilveen voi liittyä millä tahansa laitteella.

Vaikka julkisissa pilvipalveluissa on paljon hyviä ominaisuuksia, on niissä silti muutamia ongelmiakin esimerkiksi tietoturvan näkökulmasta. Käyttäjällä ei yleensä ole tietoa datan maantieteellisestä säilytyspaikasta. Tiedot saattavat päätyä maahan, jossa yksityisyyden lait ja määräykset eivät ole samalla tasolla kuin kotimaassa. Kuten aiemmin jo mainittiinkin, data on saatavilla mistä tahansa internetyhteyden välityksellä. Tämä avaa hyökkääjälle mahdollisuuden hyökätä mistä tahansa, joka tarkoittaa suoraan kas-

vavaa tietoturvariskiä. Tästä syystä arkaluonteista materiaalia ei kannata tallentaa julkiseen pilveen.

Julkisen pilven houkuttavuutta yrityksissä lisää aloituksen helppous ja alkuinvestoinnin edullisuus, mutta kannattaa pitää mielessä, että kulut kasvavat käyttäjämäärien ja toteutuneen käytön mukaan.

Yritysten näkökulmasta haittapuolena on, ettei julkisen pilven tuotteilla ole perinteisen toteutustavan mukaista hallittavuutta, koska palveluntarjoaja omistaa käytettävät laitteet ja ohjelmistot. Julkinen pilvi on myös suoraan sidonnainen käytettävissä olevan internet yhteyden nopeuteen, sillä isoja tiedostomääriä käsiteltäessä yksityisen pilven toteutusmalli palvelee julkista pilveä paremmin.

Julkiset pilvipalvelut ovat tekniseltä toteutukseltaan yleensä melkein identtiset yksityisen pilven kanssa. Erot julkisen ja yksityisen välillä tulevat palvelun rajoitettavuudesta ja käyttäjämääristä. Julkinen pilvi on avoinna kaikille kaikkialta, taas yksityinen pilvi on rajattu esimerkiksi yrityksen sisäverkkoon tai toimialueeseen. Pilvipalveluista puhuttaessa tarkoitetaan yleensä juuri julkista pilveä.

4.2 Yksityinen pilvi

Yksityinen pilvi (Private cloud) on pilvimalli, jossa on julkisen pilven kaltaiset edut, kuten skaalautuvuus ja itsepalvelu. Yksityiset pilvimallit tarjoavat virtualisoitua laskentatehoa ja tallennuskapasiteettia samaan tapaan kuin julkinen pilvi. Yksityinen pilvi voidaan ostaa yritykselle virtualisoituna palveluntarjoajalta, jolloin fyysiset laitteistot sijaitsevat palveluntarjoajan konesaleissa. Tässä tapauksessa laskutus tapahtuu julkisen pilven tapaan kuukausilaskutteisena tai toteutuneen käytön mukaan.

Yksityinen pilvi voidaan myös perustaa omaan konesaliin olemassa olevan ympäristön rinnalle. Tällöin olemassa olevien laitteistojen laskentateho saadaan käytettyä hyödyksi, eikä pilvi ei ole sidonnainen internetyhteyden katkeamiseen. Yksityinen pilvi on tällöin yritykselle kertaluontoinen investointi, koska pilveä ylläpidetään olemassa olevilla laitteistoilla, ei siitä aiheudu julkisen pilvimallin kaltaisia kuukausi- tai siirtomaksuja.

Yritys usein hallitsee pilveä itse, joten malli mahdollistaa laitteiston ja ohjelmistojen hallinnan ja räätälöinnin julkista pilveä paremmin. Kustannuksia aiheutuu, kun joudutaan ostamaan laitteita ja palkkaamaan henkilökuntaa laitteiden ylläpitoon.

Palveluntarjoajalta ostettu yksityinen pilvi on monesti yrityksen tiloihin rakennettua pilveä toimintavarmempi, koska fyysiset laitteet sijaitsevat useammassa eri palvelinsalissa. Palveluntarjoajalta ostettu palvelu on suojassa esimerkiksi tulipaloilta, vesivahingoilta ja virtapiikeiltä. Mikäli yksityinen pilvi rakennetaan yrityksen palvelimille, suurimmat riskit liittyvät fyysisten laitteiden vaurioihin.

Kuva 7, Yksityistä pilveä kuvataan usein lukolla viitaten parempaan tietoturvaan.

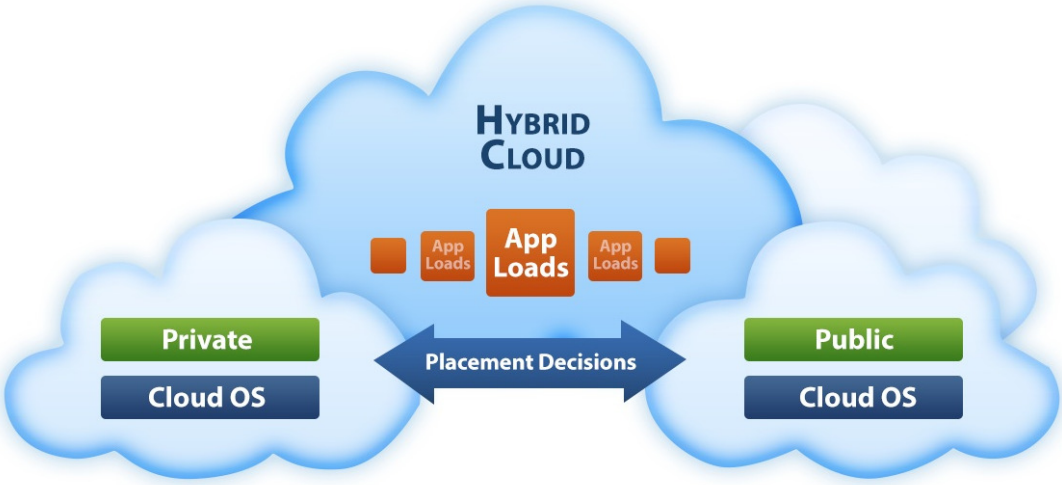
Toisin kuin julkisissa pilvissä, jotka tarjoavat palveluita useille organisaatioille ja yrityksille, yksityinen pilvi on rajattu ainoastaan yhdelle toimijalle. Tämä parantaa tietoturvaa oleellisesti, koska hyökkääjän pilveen pääsemiseksi tarvitsee ensin päästä yrityksen tai organisaation verkkoon ja sitten vasta pilveen.

4.3 Hybridi pilvi

Hybridipilvi (hybrid cloud) on pilvipalvelutyyppi, joka rakentuu kahdesta tai useammasta erillisestä pilvi-infrastruktuurista. Hybridipilvi käyttää yleensä yrityksen tiloissa sijaitsevaa yksityistä pilveä ja palveluntarjoajan konesaleissa pyörivää julkista pilveä yhtenä kokonaisuutena. Hybridipilvessä pyritään yhdistämään julkisen pilven kustannustehokkuus ja joustavuus yksityisen pilven parempaan tietoturvaan ja hallittavuuteen.

Hybridipilvi on parhaimmillaan yritysmaailman nopeasti muuttuvien laskentateho tarpeiden kanssa. Esimerkiksi yrityksen taloushallinnon tai muiden isojen ajojen aikana laskentatehon tarve kasvaa moninkertaiseksi verrattuna normaali tilanteeseen. Tällöin on halvempaa ostaa tarvittava laskentateho palveluntarjoajalta, eikä lähteä investoimaan omaan rautaan, josta jäisi normaalitilanteessa huomattavasti kapasiteettia käyttämättä.

Myös ohjelmistoa voidaan pyörittää yksityisessä pilvessä, mutta tilapäisiä kuormituspiikkejä varten voidaan hankkia lisää laskentatehoa julkisen pilven puolelta. Samalla tapaa voidaan toimia myös tietojen tallennuksen kanssa: tietoturvan kannalta kriittiset tiedot säilytetään yksityisessä pilvessä ja vähemmän kriittiset julkisessa pilvessä. [2.]



Kuva 8, Hybridipilvellä yhdistetään julkisen ja yksityisen pilven edut.

5 Fyysisen alustan valinta

Työn tarkoituksena oli tehdä toimiva pilvipalveluympäristö mahdollisimman pienillä kustannuksilla, joten suunnitteluvaiheessa fyysisen raudan valinta oli keskeisessä roolissa.

Alustavalintaa tehdessä vertailtiin lähinnä seuraavia asioita:

- hintaa
- ytimien määrää

- prosessorin kellotaajuutta
- keskusmuistin määrää
- keskusmuistin nopeutta
- virrankulutusta.

Vaikka ODROID-XU4-alustaa ei ollut vielä projektin alussa julkaistu, otettiin se kuitenkin vertailuun mukaan sen kiinnostavien ominaisuuksiensa vuoksi. Taulukossa 1 vertaillaan ARM-pohjaisten minitietokoneiden teknisiä ominaisuuksia.

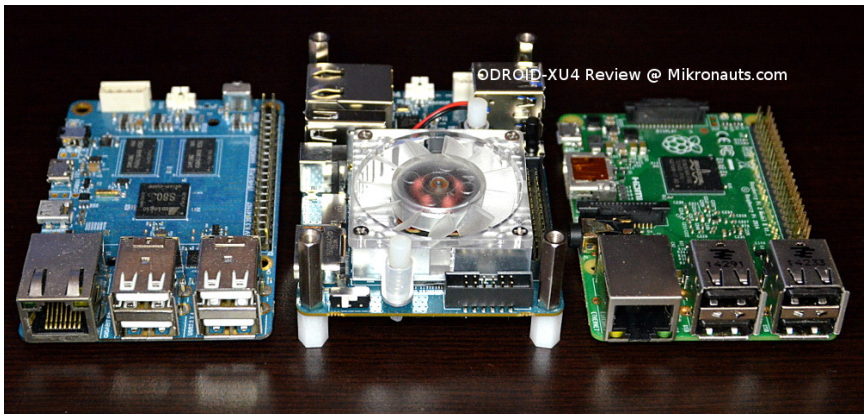
Taulukko 1. Eri alustojen tekniset tiedot [4.]:

	Banana Pro	ODROID	Raspberry Pi	Raspberry Pi	ODROID
		C1	**Model B+**	**2 Model B**	**XU4**
Processor	Allwinner A20	Amlogic S805	BCM2835	BCM2836	Samsung Exynos5422
Architecture	ARMv7	ARMv7	ARMv6	ARMv7	ARM v7
Cores	2	4	1	4	8
Clock Speed	912MHz*	1500Mhz	700Mhz	900Mhz	2.0/1.4Ghz
BogoMips	3824	timer	698	timer	timer
GPU	Mali 400	Mali 450	VideoCore IV	VideoCore IV	Mali T628
Memory Size	1GB	1GB	512MB	1GB	2GB
Memory Type	DDR3	DDR3	SDRAM	LPDDR2	LPDDR3
Mrmoty Mhz	432	400	400	450	933
GPIO pins	40	40	40	40	30+12
LCD Flex Socket	LVDS	–	DSI	DSI	–
Ethernet	10/100/1000*	10/100/1000*	10/100	10/100	10/100/1000
Controller	Allwinner A20	Amlogic S805	LAN9514	LAN9514	Realtek RTL8153
eMMC	–	eMMC socket	–	–	eMMC 5.0
SD	uSD	UHS-1 uSD	uSD	uSD	uSD
HDMI out	YES	YES	YES	YES	YES
USB 3.0	–	–	–	–	2
USB Host	2	4	4	4	1
MSRP	**$45.00**	**$35**	**$35.00**	**$35.00**	**$75.00**

Taulukko 2. Virrankulutukset [5.]

Malli	Maksimi (W)	Keskimääräinen (W)
RPi B+	1,2	1,15
RPi 2 B	2,25	1,55
Banana Pi	2	1,25
Banana Pro	2,3	1,62
Odroid-C1	2,3	1,62

Näiden taulukoiden perusteella alustan valinta olisi ollut ODROID. Vaihtoehtojen laajemman tarkastelun perusteella todettiin, että Raspberry Pi:n kehitysyhteisö on moninkertainen verrattuna kilpailijoihin. Raspberryn suosion ja laajan kehitysyhteisön takia se on tuettu ohjelmistokehittäjien toimesta, jolloin yhteensopivuusongelmat ovat vähäisempiä kuin muilla vertailussa olleilla. Teimme asiakkaan kanssa päätöksen käyttää Raspberry Pi:tä alustana rakennusvaiheessa ja mahdollisesti päivittää rauta myöhemmin nopeampaan.



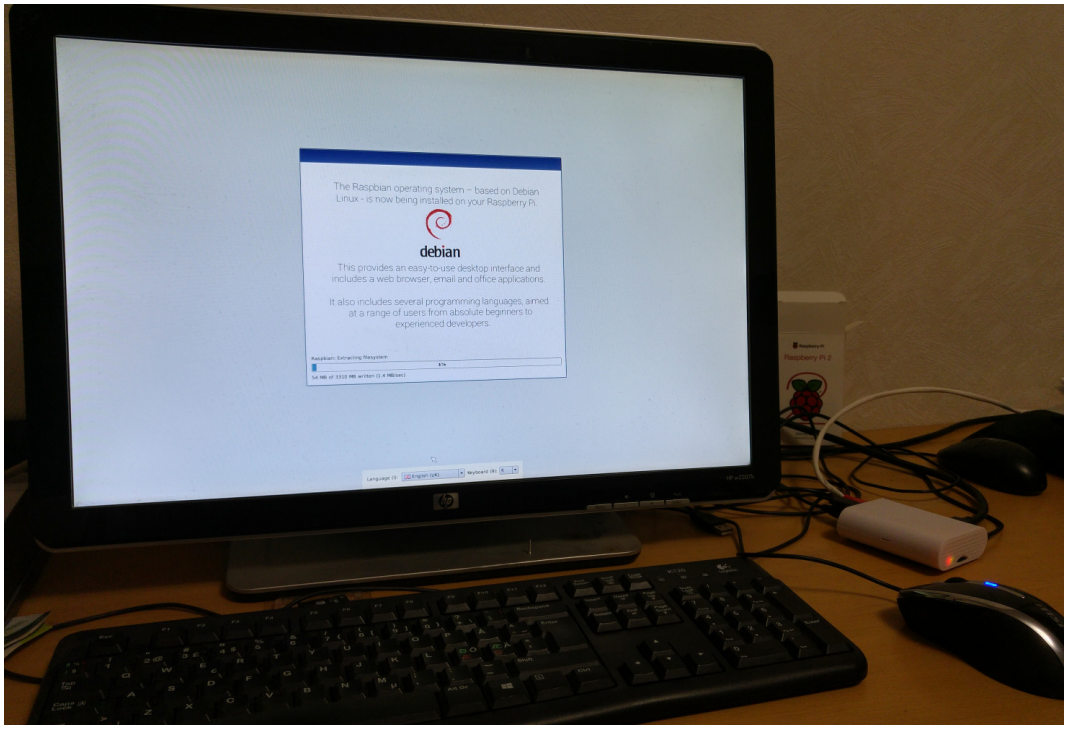
Kuva 9, Vasemmalla ODROID-C1, keskellä ODROID-XU4 ja oikealla Raspberry Pi 2.

5.1 Raspberry Pi

Raspberry Pi (RPi) on luottokortin kokoinen yhden piirilevyn tietokone, joka pystyy käytännössä samaan kuin normaali pöytäkonekin. Laitteen kehityksestä vastaa Raspberry Pi Foundation, joka on opetusjärjestö tavoitteenaan kannustaa ihmisiä tietojenkäsittelytieteen ja siihen liittyvien aiheiden pariin. Raspberry Pi Foundation saa tukea muun muassa Cambridgen yliopistolta ja piirivalmistaja Broadcomilta.

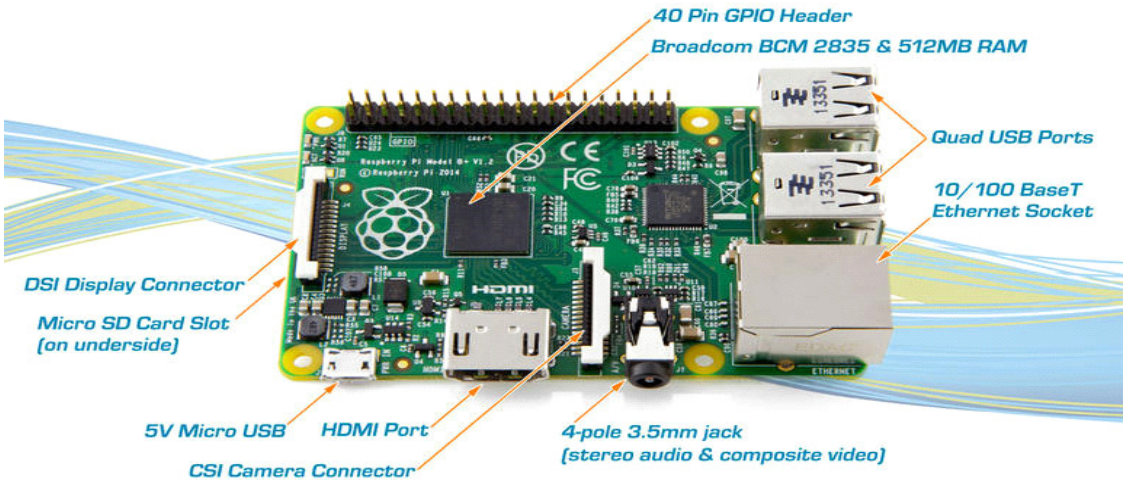
Ensimmäinen laite julkaistiin 29. helmikuuta 2012, mallinimellä A. Kolme vuotta myöhemmin 2015 säätiö julkaisi Raspberry Pi 2:n, joka on ulkomitoiltaan ja kiinnityspaikoiltaan täysin A-version kaltainen, tarjoten kuitenkin parannellun suorittimen sekä kaksinkertaistetun keskusmuistin.

Laitteiston kokoonpano perustuu Broadcomin valmistamaan järjestelmäpiiriin (BCM2836), joka sisältää suorittimen, muistin ja integroidun grafiikkapiirin.



Kuva 10, Rasperry Pi:n käyttöönottoon vaadittava laitteisto.

RPi:n käyttöönottoon tarvittavat liitännät ovat: HDMI-liitännän avulla voidaan lisätä mikä tahansa HDMI:tä tukeva näyttö. Syöttölaitteina toimivat melkein kaikki USB-liitännäiset näppäimistöt ja hiiret. Internetyhteys saadaan RJ45-portista ja käyttövirta tulee micro-USB-liitännällä. [3.]



Kuva 11, Raspberry Pi:n komponenttisijoittelu.

5.2 Käyttöjärjestelmä

Käyttöjärjestelmävaihtoehtoja olivat Raspian, Ubuntu Mate, Windows 10 IoT Core, Android, Arch Linux ARM, RISC OS tai openSUSE. Näistä käyttöjärjestelmistä valikoitui Raspian ja Ubuntu Mate, joiden välillä tehtiin tarkempaa vertailua. Valintaperusteena oli lähinnä käyttöjärjestelmän suosio ja yhteensopivuus projektiin. Näistä kahdesta käyttöjärjestelmäksi valittiin Raspian.

Raspian on Debian Linux-jakelupakettiin pohjautuva ilmainen käyttöjärjestelmä, joka on optimoitu nimenomaan Raspberry Pi:n laitteistolle. Optimoinnin avulla Raspian käyttää resursseja sieltä, missä niitä on. Raspian on Raspberry Pi Foundationin virallisesti tukema käyttöjärjestelmä. Se ei kuitenkaan ole Raspberry Pi Foundationin ylläpitämä, vaan kehityksestä vastaa erittäin aktiivisten harrastajien yhteisö.

Tässä työssä käytettiin käyttöjärjestelmän asennuksessa NOOBS-ohjelmistoa, joka on tehty helpottamaan käyttöjärjestelmien asennusta Raspberry Pi alustalle.

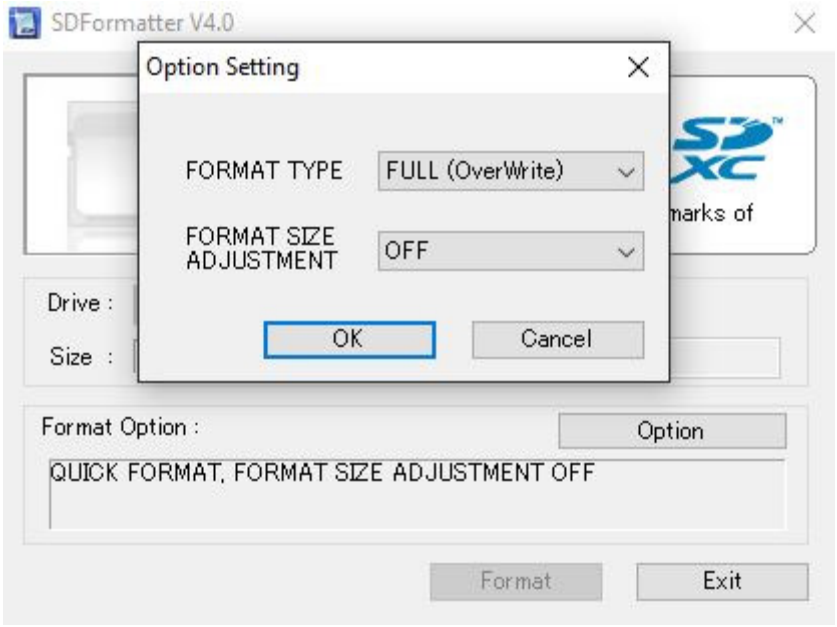
6 IaaS-pilvipalvelun toteutus

Tässä luvussa käydään läpi pilvipalvelun asennus ja tarvittavat asetukset.

6.1 Raspian-käyttöjärjestelmän asennus

Raspian vaatii toimiakseen vähintään neljän gigatavun micro-SD muistikortin. Tässä työssä valitsimme käyttöömme Kingstonin 32 gigatavun microSDHC- muistikortin nopeusluokalla UHS-1. Valmistajan ilmoittama lukunopeus kortille on 45Mt/s ja kirjoitus 10Mt/s, joka riittää RPi:n laskentateholle mainiosti.

SD-kortin alustamisessa käytettiin SD-Formatter ohjelmaa, jota Raspberry Pi yhteisö suosittelee. SD-Formatter ladattiin SD Association yhtiön kotisivuilta [8]. Kortin alustaminen on suositeltavaa, vaikka kyseessä olisikin uusi kortti. Kortin alustuksella varmistetaan, ettei kortilla ole ylimääräistä dataa ja että tiedostojärjestelmämuodoksi on määritelty (FAT32).



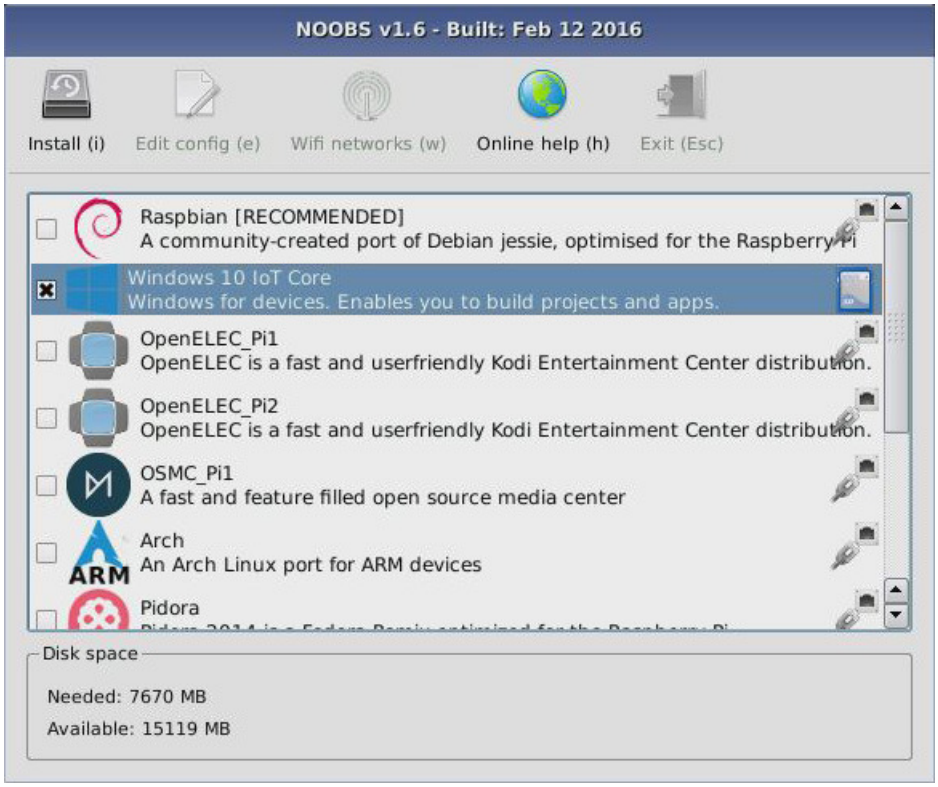
Kuva 12, SD Formatterin käyttöliittymä.

SD Formatterin käyttö on yksinkertaista. Ohjelma tunnistaa kortin automaattisesti ja näyttää sen tiedot. Pika-alustus olisi riittänyt, mutta käytimme varmuuden vuoksi vahvinta mahdollista alustusta.

Kortin alustuksen jälkeen ladattiin NOOBS-ohjelmisto. NOOBS-ohjelmistoa on tarjolla kahdeksaa eri pakettivaihtoehtoa, mikä saattaa sekoittaa aloittelevaa käyttäjää. Tässä työssä käytimme NOOBS:in ilman internetyhteyttä asennettavaa kokonaista pakettia. Tämän jälkeen zip-muotoinen tiedosto purettiin ja siirrettiin alustetulle SD-kortille. SD-kortin kirjoitusnopeus oli valmistajan ilmoittamaa hitaampi ja se esitetään kuvassa 10.

Kuva 13, Hieman ilmoitettua huonompi siirtonopeus.

Siirron jälkeen SD-kortti ja tarvittavat johdot yhdistettiin Raspberry Pi:hin. Virtajohto kytkettiin viimeisenä. Rasperry Pi:ssä ei ole virtakytkintä, joten laite siis käynnistyy välittömästi virtajohdon kytkemisen jälkeen. Laitteen käynnistyttyä valittiin asennettava käyttöjärjestelmä, jonka käyttöliittymä on esitetty kuvassa 11.



Kuva 14, NOOBS-ohjelmiston käyttöjärjestelmävalinta.

Käyttöjärjestelmiä on mahdollista asentaa useampia samanaikaisesti, mutta tässä työssä siihen ei ollut tarvetta.

Kun käyttöjärjestelmä oli asentunut, aukesi ruudulle Raspberry Pi:n yleiset asetukset. Tämän ikkunan voi ohittaa tässä vaiheessa, koska sen voi avata myöhemmin komennolla:

```
sudo raspi-config
```

6.2 Raspianin yleiset asetukset

1. Change User Password
   Vaihdettiin salasana. Tämä kannattaa tehdä heti ensimmäisen käynnistyksen yhteydessä.

2. Internationalisation options
   Raspianissa on oletuksena englanninkielinen näppäimistö, joten se vaihdettiin suomenkieliseen.

3. Expand Filesystem
   Otetaan käytettävissä oleva tallennustila kokonaisuudessaan käyttöön.

4. Enable Boot to Desktop / Scratch
   Disabloidaan työpöytä, jotta saadaan vapautettua RPi:n grafiikkapiirin resursseja prosessorille. Työpöytä saadaan terminaaliin käynnistyksen jälkeen käyntiin komennolla startx.

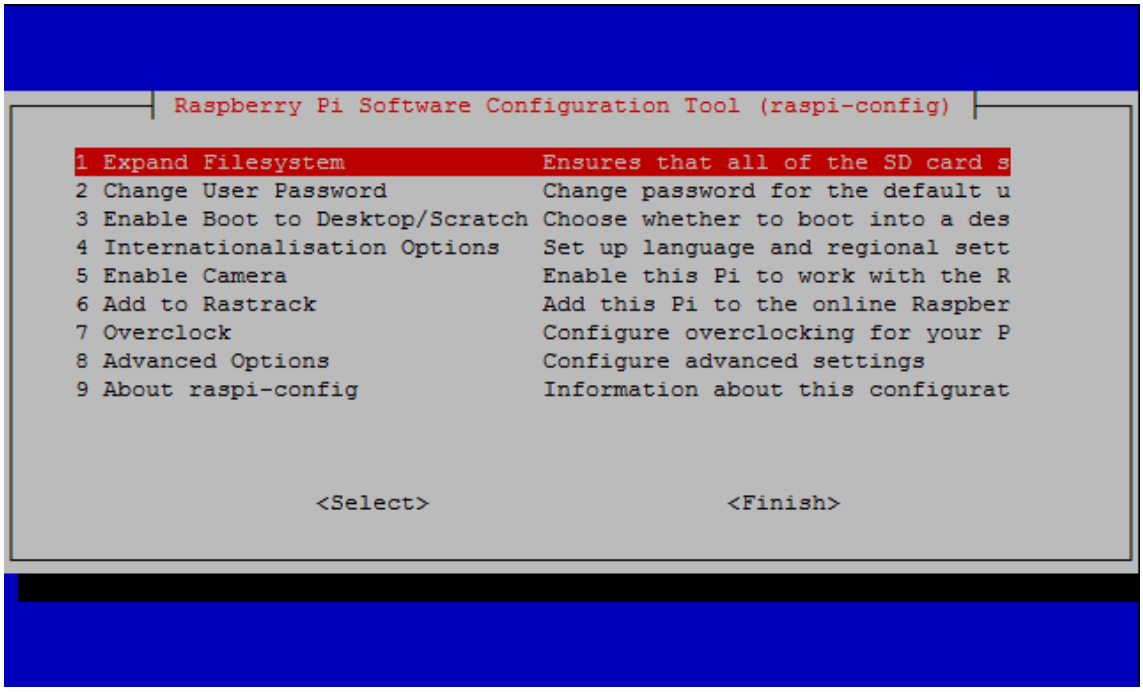
5. Overclock
   Nostettiin RPi:n kellotaajuutta asetukseen 1000MHz ARM, 500MHz core, 500MHz SDRAM, 2 overvolt.

6. Hostname
   Valitaan RPi:lle nimi joka näkyy myöhemmin mm. verkkonimenä.

7. Memory Split
   RPi:n RAM muisti on jaettu näytönohjaimen ja prosessorin kesken. Tässä työssä ei tarvita näytönohjainta, joten näytönohjaimelle jätettiin ainoastaan 16Mb ja loput vapautettiin prosessorille.

8. SSH
   Otetaan käyttöön RPi:n salattu hallintayhteys.

Asetusten määrittelyn jälkeen RPi uudelleenkäynnistetään, jotta asetukset tulevat voimaan.



Kuva 15, Raspi-config käyttöliittymä.

6.3 Verkkoasetusten määrittely

Uudelleenkäynnistyksen jälkeen tarkistetaan, onko RPi saanut noudettua itsenäisesti ip-osoitetta komennolla:

```
ifconfig
```

DHCP:lla jaeltu osoite löytyi, mutta se haluttiin määritellä staattiseksi, etteivät verkon muutokset vaikuttaisi RPi:n toimintaan. Tämä onnistui muokkaamalla interfaces-tiedostoa nano-tekstieditorilla.

```
sudo nano /etc/network/interfaces
```

Verkkokortin eth0 dynaaminen osoite muutettiin staattiseksi ja ip-osoite määriteltiin seuraavasti:

```
auto lo
iface lo inet loopback
iface eth0 inet static

address 133.7.0.40
netmask 255.255.255.0
network 133.7.0.0
broadcast 133.7.0.255
gateway 133.7.0.1
```

Ctrl + O tallentaa muutokset ja Ctrl + X sulkee editorin.

Heti käyttöönoton jälkeen törmättiin nimiristiriita virheeseen:

```
sudo: unable to resolve host raspi
```

Virhe toistui jokaisen komennon jälkeen.
Syy oli siinä, ettei RPi:n normaaleiden käyttöönotto ohjeiden mukaan tehty ip-määrittely toiminut ipv6-ympäristössä, joten RPi:n määrittelytiedostoon oli lisättävä vielä ipv6-määrittelyt ja isäntänimi:

```
sudo nano /etc/hosts

127.0.0.1       localhost
::1             localhost ip6-localhost ip6-loopback
fe00::0         ip6-localnet
ff00::0         ip6-mcastprefix
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
127.0.1.1       WPraspi
```

Muutosten jälkeen uudelleen käynnistettiin verkkokortin palvelu:

```
sudo services networking restart
```

Muutosten voimaantulo tarkistettiin komennolla:

```
ifconfig
eth0      Link encap:Ethernet  HWaddr
b8:37:eb:40:f8:28
inet addr:133.7.0.40  Bcast: 133.7.0.255
Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:28632 errors:0 dropped:0 overruns:0
frame:0
TX packets:23820 errors:0 dropped:0 overruns:0 car-
rier:0
RX bytes:12901527 (12.3 MiB)  TX bytes:4646116 (4.4
MiB)
```

Verkon käyttöönoton jälkeen päivitettiin käyttöjärjestelmä ja ohjelmistot uusimpiin versioihin:

```
sudo apt-get update -y
```

```
sudo apt-get upgrade -y
```

Komennot lataavat ja asentavat päivityspaketit, -y vivulla hyväksytään automaattisesti mahdolliset kysymykset.

Tietoturvasyistä on järkevää luoda RPi:lle käyttäjä, jota ei vakiokokoonpanossa käytetä. Uuden käyttäjän luominen onnistuu seuraavilla komennoilla:

```
groups
```

Komento näyttää aktiiviset ryhmät, joilla on käyttöoikeuksia:

```
pi adm dialout cdrom sudo audio video plugdev games
users input netdev gpio i2c spi
```

Seuraavalla komennolla annetaan täydet oikeudet resursseihin ja luodaan käyttäjä vee:

```
sudo useradd -m -G
adm,dialout,cdrom,sudo,audio,video,plugdev,games,use
rs,input,netdev,gpio,i2c,spi vee
```

Määritellään salasana käyttäjälle vee:

```
sudo passwd vee
```

Seuraavaksi poistetaan käyttäjä pi, joka on RPi:lle luotu vakiokäyttäjä. Koodiin oli päästetty pätkä huumoria:

```
sudo deluser --remove-all-files pi

We trust you have received the usual lecture from
the local System Administrator. It usually boils
down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
```

```
#3) With great power comes great responsibility.
```

6.4 Avainpariautentikointi SSH:lle

Etähallinnan tietoturvan parantamiseksi otetaan avainpariautentikointi SSH:lle käyttöön. Muodostetaan ”public” ja ”private” avainparit:

```
ssh-keygen -b 4096
```

Ilman –b vipua komento muodostaisi 2048-bittisen avaimen. –b-vivulla saadaan kasvatettua avaimen salaus 4096-bittiseksi. RPi:n laskentatehoilla avaimen muodostus kestää useita minuutteja.

```
Generating public/private rsa key pair.
(/home/pi/.ssh/id_rsa): avain
Your identification has been saved in avain.
Your public key has been saved in avain.pub.
The key fingerprint is:
The key's randomart image is:
+--[ RSA 4096]----+
|    +     .      |
|    +    o       |
|    .   . .  .   |
|      .  .o . .  |
| .      +S.o.. ..|
| +     o o o=. oo|
|     . o.* .. o  |
|      . . o .    |
|      o     E    |
+-----------------+
```

Avaimen luonnin jälkeen otetaan WinSCP-ohjelmalla FTP-yhteys RPi:hin ja siirretään juuri luotu avain tiedostohallintaan käytettävälle tietokoneelle. Tämän jälkeen kotihakemistoon luodaan kansio avaimelle:

```
sudo mkdir .ssh
```

Siirretään public avain sallittujen avainten listalle:

```
sudo mv avain.pub .ssh/authorized_keys
```

Määritellään tiedoston käyttöoikeudet:

```
sudo chown -R vee:vee .ssh
sudo chmod 700 .ssh
sudo chmod 600 .ssh/authorized_keys
```

Tämän jälkeen SSH kirjautuminen tapahtuu annetun “passphasen” avulla.

6.5 SSH salasana autentikoinnin ja root kirjautumisen disablointi.

Avataan SSH konfiguraatiotiedosto:

```
sudo nano /etc/ssh/sshd_config
```

Otetaan salasana-autentikointi ja root kirjautuminen pois päältä:

```
PasswordAuthentication no
PermitRootLogin no
```

Käynnistetään ssh-palvelu uudelleen komennolla:

```
sudo service ssh restart
```

6.6 Palomuurin konfigurointi

Ensin katsotaan aktiiviset säännöt. Tässä vaiheessa kaikki pitäisi olla sallittuna.

```
sudo iptables -L
```

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Tehdään tiedosto palomuurisäännöille komennolla:

```
sudo nano /etc/iptables.firewall.rules
```

Lisätään tiedostoon haluttavat säännöt:

```
*filter
#  Sallitaan kaikki (lo0) liikenne ja estetään lii-
kenne jota ei käytetä 127/8 lo0
-A INPUT -i lo -j ACCEPT
-A INPUT -d 127.0.0.0/8 -j REJECT

#  Accept all established inbound connections
-A INPUT -m state --state ESTABLISHED,RELATED -j AC-
CEPT

#  Allow all outbound traffic
-A OUTPUT -j ACCEPT

#  Allow HTTP and HTTPS connections from anywhere
-A INPUT -p tcp --dport 80 -j ACCEPT
-A INPUT -p tcp --dport 443 -j ACCEPT

# Allows SMTP access
-A INPUT -p tcp --dport 25 -j ACCEPT
-A INPUT -p tcp --dport 465 -j ACCEPT
-A INPUT -p tcp --dport 587 -j ACCEPT
```

```
# Allows pop and pops connections
# -A INPUT -p tcp --dport 110 -j ACCEPT
# -A INPUT -p tcp --dport 995 -j ACCEPT

# Allows imap and imaps connections
-A INPUT -p tcp --dport 143 -j ACCEPT
-A INPUT -p tcp --dport 993 -j ACCEPT

#  Allow SSH connections
#  The -dport number should be the same port number
you set in sshd_config
-A INPUT -p tcp -m state --state NEW --dport 22 -j
ACCEPT
#  Allow ping
-A INPUT -p icmp --icmp-type echo-request -j ACCEPT

#  Log iptables denied calls
-A INPUT -m limit --limit 5/min -j LOG --log-prefix
"iptables denied: " --log-level 7

#  Drop all other inbound
-A INPUT -j DROP
-A FORWARD -j DROP

COMMIT
```

Otetaan palomuurisäännöt käyttöön komennolla:

```
sudo iptables-restore < /etc/iptables.firewall.rules
```

Palomuurisääntöjen käyttöönoton jälkeen aktiivisten sääntöjen lista näyttää tältä:

```
Chain INPUT (policy ACCEPT)
target     prot opt source     destination
ACCEPT     all  --  anywhere   anywhere
```

```
REJECT     all  --  anywhere   loopback/8  reject-
with icmp-port-unreachable
ACCEPT     all  --  anywhere   anywhere    state RE-
LATED,ESTABLISHED
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:http
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:https
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:smtp
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:ssmtp
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:submission
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:imap2
ACCEPT     tcp  --  anywhere   anywhere    tcp
dpt:imaps
ACCEPT     tcp  --  anywhere   anywhere    state NEW
tcp dpt:ssh
ACCEPT     icmp --  anywhere   anywhere    icmp
echo-request
LOG        all  --  anywhere   anywhere    limit:
avg 5/min burst 5 LOG level debug prefix "iptables
denied:"
DROP       all  --  anywhere   anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source     destination
DROP       all  --  anywhere   anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source     destination
ACCEPT     all  --  anywhere   anywhere
```

Nämä säännöt eivät aktivoidu automaattisesti RPi:n käynnistyksessä, joten teemme niille käynnistysskriptin:

```
sudo nano /etc/network/if-pre-up.d/firewall
```

Lisätään rivit:

```
#!/bin/sh
/sbin/iptables-restore <
/etc/iptables.firewall.rules
```

Määritellään käyttöoikeudet:

```
sudo chmod +x /etc/network/if-pre-up.d/firewall
```

6.7 Fail2Ban sanakirjahyökkäyksiä vastaan

```
sudo apt-get install fail2ban
```

Fail2Ban seuraa epäonnistuneita kirjautumisyrityksiä ja estää ne verkkotasolla lisäämällä palomuurisääntöihin väliaikaisen eston hyökkääjän IP-osoitteelle ja kirjaa ne lokitiedostoon. Loki tallennetaan OwnCloudiin, jotta se on helposti seurattavissa. Vakioasetuksilla Fail2Ban suojaa SSH-yhteyksiä, mutta se kykenee myös muiden protokollien seuraamiseen ja estämiseen. Tuettuja protokollia ovat SSH:n lisäksi mm. HTTP ja SMTP.

6.8 nginx-asennus

WordPress ei pysty toimimaan itsenäisesti, vaan se tarvitsee toimiakseen muita ohjelmia. Ensin asennetaan nginx:

```
sudo apt-get install nginx -y
```

Asentamisen jälkeen tehdään muutoksia nginxin konfiguraatioon RPi:n rajoituksista johtuen:

```
sudo nano /etc/nginx/nginx.conf
```

Lisätään raja-arvot:

```
http{
block
client_body_buffer_size 10K;
client_header_buffer_size 1k;
client_max_body_size 8m;
large_client_header_buffers 4 16k;
}
```

Konfiguraatiotiedostosta löytyi oleellisesti tietoturvaa parantava kohta, joka otettiin käyttöön:

```
server_tokens off
```

Tämän jälkeen nginx ei enää raportoi sivuston käyttäjälle käytössä olevaa nginx-versiota. Mikäli versio näytetään, pystyy mahdollinen hyökkääjä etsimään käytössä olevan version tietoturva-aukot ja hyökätä niiden avulla.

ngixissä on Gzip ominaisuus, jolla pakataan tieto ennen sen lähettämistä verkkoon. Tämä vähentää liikennettä, joka taas nopeuttaa sivujen latautumista. Haittapuolena on, että ominaisuus käyttää RPi:n rajoitettuja resursseja.

Gzip asetukset:

```
    gzip on;
    gzip_disable "msie6";

gzip_min_length   1100;
gzip_vary         on;
gzip_proxied      any;
```

```
gzip_buffers      16 8k;
gzip_comp_level   6;
gzip_http_version 1.1;
gzip_types        text/plain text/css applci-
ation/json application/x-javascript text/xml appli-
cation/xml application/rss+xml text/javascript imag-
es/svg+xml application/x-font-ttf font/opentype
                  application/vnd.ms-fontobject;
```

Yllä näkyvässä koodissa otetaan Gzip käyttöön. Sille kerrotaan, että pakataan ainoastaan isompia tiedostoja, valitaan välimuistit ja kerrotaan, minkä tyyppisiä tiedostoja pakataan. gzip_comp_level-arvoa muuttamalla säädetään pakkausastetta. Arvoja voi syöttää väliltä 1-9. Gzip vie RPi:n prosessoritehoja, joten valitsimme normaalia hieman pienemmän arvon 6.

Samassa yhteydessä on aihetta parantaa nginxin tietoturvaa mm. palvelunestohyökkäyksiä vastaan lisäämällä seuraavat rivit:

```
client_header_timeout 10;
client_body_timeout   10;
keepalive_timeout     10 10;
send_timeout          10;
```

nginxin toiminta varmennetaan avaamalla selaimella RPi:lle määritelty ip-osoite.

Kuva 16, Onnistunut nginx-asennus.

6.9 PHP:n ja MySQL:n asennus

```
sudo apt-get install php5-mysql php5-cli php5-curl
php5-gd php5-fpm -y
```

Asennetaan MySQL server ja PHP-paketteja

```
sudo apt-get install php5-mcrypt php-apc mysql-
server -y
```

6.10 SQL-serverin käyttöönotto ja suojausasetukset:

```
sudo mysql_secure_installation
```

MySQL-asennuksessa syötettiin käytettävät tunnukset ja oletustietokannan nimi. Päätettiin suojausasetuksista ja poistettiin ilman tunnistetietoja tapahtuva kirjautuminen. Seuraavaksi tehtiin WordPress-käyttäjä ja määriteltiin sille tietokanta. Sen jälkeen käyttäjälle annettiin oikeudet SQL-tietokantaan:

```
mysql> CREATE USER vee@localhost IDENTIFIED BY 'sa-
lasana';
mysql> CREATE DATABASE wordpress;
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO
vee@localhost IDENTIFIED BY 'salasana';
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

WordPressin liittäminen nginx:iin

```
sudo nano /etc/nginx/sites-available/ac-tekniikka.fi
```

Tiedostoon lisätään seuraavat määrittelyt:

```
server {
```

```
        server_name www.ac-tekniikka.fi ac-
tekniikka.fi;

        access_log   /var/log/nginx/ac-
tekniikka.fi.access.log;
        error_log    /var/log/nginx/ac-
tekniikka.fi.error.log;

        root /var/www/ac-tekniikka.fi/;
        index index.php;

        location / {
                try_files $uri $uri/
/index.php?$args;
        }
        location = /favicon.ico {
                log_not_found off;
                access_log off;
        }
        location = /robots.txt {
                allow all;
                log_not_found off;
                access_log off;
        }
        Deny public access to wp-config.php
        location ~* wp-config.php {
                deny all;
        }
        location ~ \.php$ {
           try_files $uri =404;
           include fastcgi_params;
           fastcgi_pass unix:/var/run/php5-fpm.sock;
           fastcgi_split_path_info ^(.+\.php)(.*)$;
           fastcgi_param  SCRIPT_FILENAME $docu
           ment_root$fastcgi_script_name;
```

```
        }
}
```

Poistetaan oletus-sivuston linkitys ja tehdään symbolinen linkki WordPress-sivustolle:

```
unlink /etc/nginx/sites-enabled/default
ln -s /etc/nginx/sites-available/wordpress
/etc/nginx/sites-enabled/wordpress
```

Käytön nopeuttamiseksi lisätään RPi:n osoite hallintakoneen hosts-tiedostoon. Lähiverkossa käytettävän sivun DNS-ohjausten lisääminen Windows 10 koneelle:

```
%SystemRoot%\System32\drivers\etc\hosts
```

Tiedostoon lisättiin sisäverkon nimiosoite, joka antaa RPi:n ip-osoitteelle nimen. Tätä osoitetta käytettäessä liikenne tapahtuu lähiverkon sisällä ja on täten huomattavasti ulkoista osoitetta nopeampi.

```
133.7.0.40 ac-tekniikka.fi.local
```

Otetaan käyttöön vaihtoehtoinen PHP-prosessiontitapa, joka nopeuttaa PHP:n käsittelyä. Ensin avataan php5-fpm pool -konfiguraatiotiedosto:

```
nano /etc/php5/fpm/pool.d/www.conf
```

Lisätään linkitys php5-fpm.sock -asetustiedostoon:

```
listen = /var/run/php5-fpm.sock
listen = 127.0.0.1:9000
```

Rajoitetaan samanaikaiset pyynnöt:

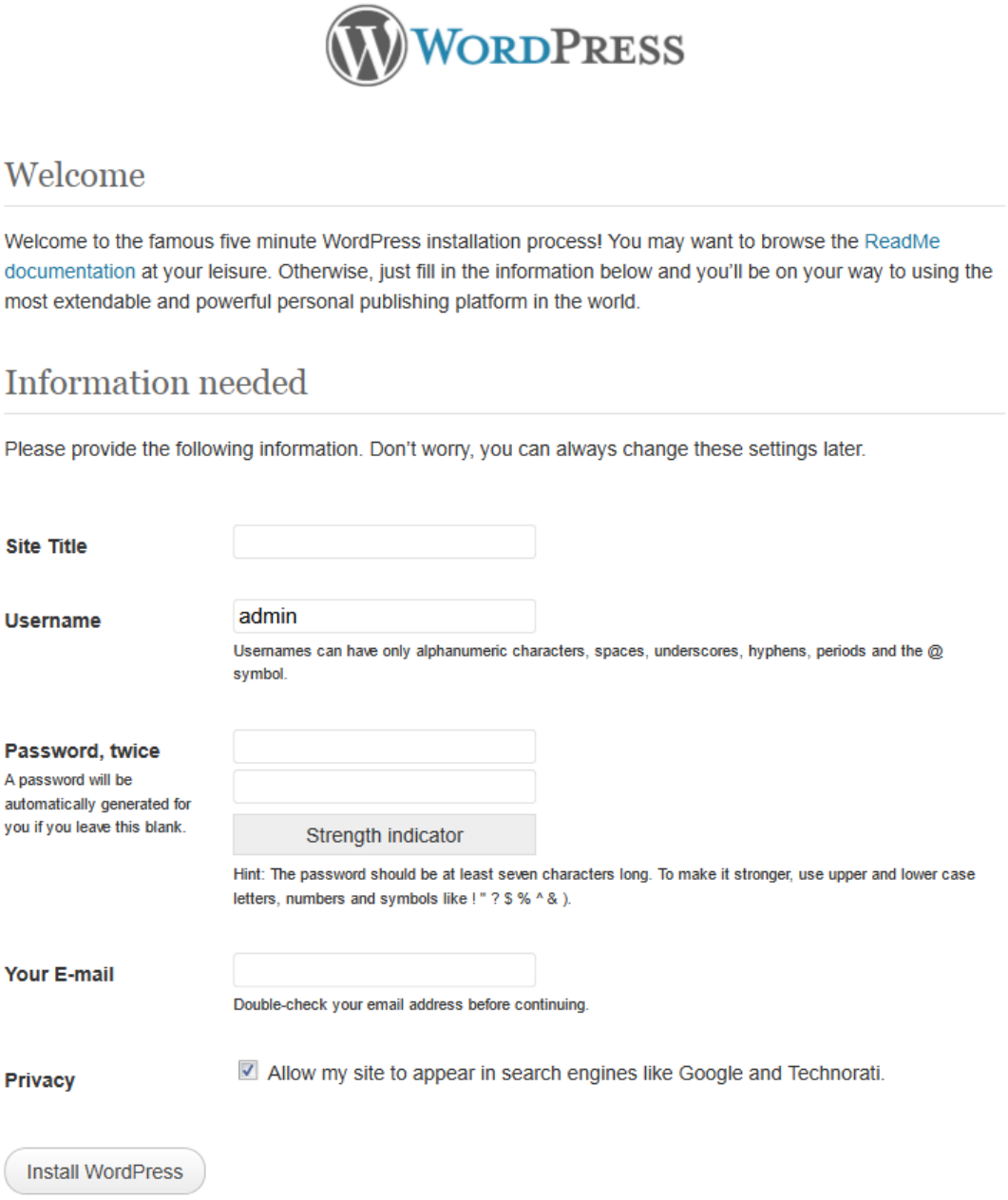
```
pm.max_requests = 200
```

Muutosten voimaantulemiseksi täytyy nginx ja PHP käynnistää uudelleen komennoilla:

```
service nginx restart
service php5-fpm restart
```

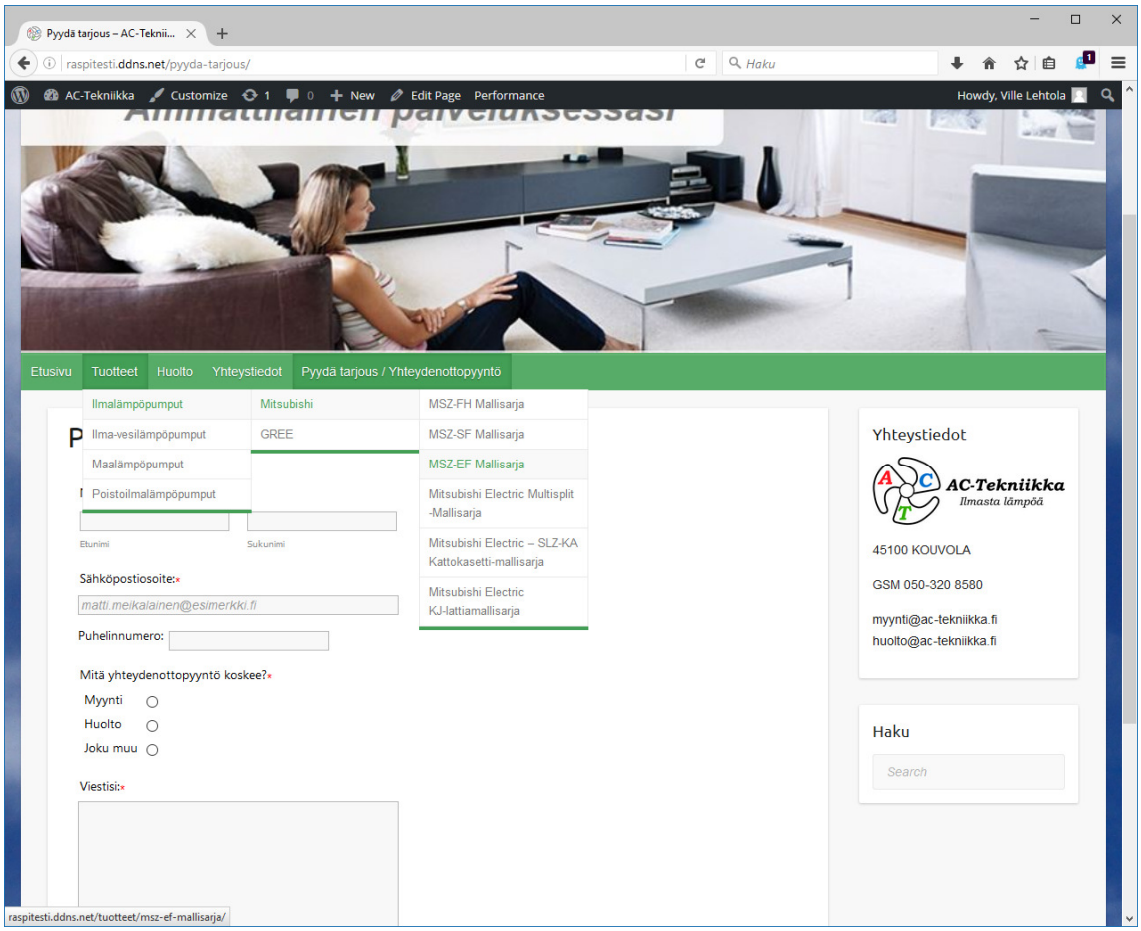
6.11 WordPress sivuston luonti

Sivuston luominen oli yksinkertainen prosessi, jossa syötettiin graafiseen ympäristöön tarvittavat tiedot.



Kuva 17, WordPress -sivuston lomake

Insinöörityön tekijän vastuulla oli luoda WordPress-sivusto ja tehdä tarvittavat muutokset teemoihin ja lisäosiin. Tässä yhteydessä luotiin sivuston alasivut, valikkorakenteet, muokattiin teeman CSS-koodia, konfiguroitiin tarvittavat lisäosat ja annettiin käytön opastusta asiakkaan suuntaan.



Kuva 18, WordPress -sivusto testiympäristössä.

Asiaakkaalle tehtiin myös kirjalliset ohjeet sivuston ylläpitoa varten, jotka löytyvät liitteestä 1.

6.12 OwnCloud-esivalmistelut

Ennen OwnCloudin asennusta alustettiin levytila, joka otettiin OwnCloudin käyttöön. Levyn alustus ja järjestelmään liittäminen on suoraviivainen toimenpide. Levyn formatointiin käytettiin Raspian paketin mukana tulevaa osiointityökalua nimeltään parted.

```
sudo parted
```

Komennolla avataan parted-työkalu.

```
print all
```

Komennolla listataan kaikki RPi:hin yhdistetyt levyasemat. Tässä tapauksessa valittavana on ainoastaan järjestelmälle varattu `/dev/mmcblk0` ja juuri yhdistämämme ulkoinen levy `/dev/sda`. Valitaan ulkoinen levy komennolla:

```
select /dev/sda
```

Varmistetaan, että valinta on onnistunut komennolla:

```
print
```

Komento näyttää valitun aseman tiedot:

```
Model: Drive_20 (scsi)
Disk /dev/sda: 2TB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Number Start   End     Size  Type File system  Flags
 1     32.3kB  2TB     2TB   primary fat32     boot
```

Tiedoista näkee, että levyn osiointityyppi on msdos. Tämän tyyppinen osiointi aiheuttaisi myöhemmin ongelmia, joten se vaihdettiin toiseen komennolla:

```
mklabel gpt
```

Levyn tietoihin muutettiin onnistuneesti osiointityyppi:

```
Partition Table: gpt
```

Seuraavaksi määritellään tiedostojärjestelmä käyttämään ext4-muotoa. Ensin tehdään liitospiste, eli kansio, joka osoittaa ulkoiselle levylle. Kansion nimeäminen vaikuttaa myöhemmin käytettävään OwnCloud -jakonimeen. Tehdään kansio komennolla:

```
sudo mkdir /data
```

Kansion luomisen jälkeen tehdään kansion liitospiste:

```
sudo mount /dev/sda1 /data
```

Levy on oletuksena saatavilla ainoastaan root-käyttäjälle, joten lisätään muille käyttäjille käyttöoikeudet:

```
sudo chgrp -R users /data
sudo chmod -R g+w /data
```

Levy on käyttövalmis, mutta teemme vielä pientä hienosäätöä, jotta levy on saatavilla automaattisesti uudelleenkäynnistyksen jälkeen.

```
ls -l /dev/disk/by-uuid/
```

Komennolla saadaan UUID näkyviin, jota käytetään levyn liittämisessä.

```
UUID=cd346d1c-f3be-4a4d-84ef-64a05eaacc82
```

Avataan fstab:

```
sudo nano /etc/fstab
```

Määrityksiin lisätään levyn UUID-koodi, liitospiste, tiedostojärjestelmä ja Raspianin vaatimat levymäärittelyt:

```
UUID=cd346d1c-f3be-4a4d-84ef-64a05eaacc82
/data   ext4    defaults,nofail      0       2
```

Levyn automaattinen liittäminen on otettu käyttöön. Seuraavaksi testataan:

```
sudo reboot
```

Katsotaan, onko levy liittynyt automaattisesti:

```
df -h
```

Onnistunut liittäminen näkyy listauksessa seuraavasti:

```
/dev/sda1       1.9T  20.9M  1.7T   1% /data
```

6.13 OwnCloudin asennus ja konfigurointi

Asiakas halusi yksityisen pilven, joka olisi mahdollisimman helposti käytettävissä. Päädyttiin käyttämään olemassa olevaa verkko-osoitetta, ettei ylimääräisiä osoitteita tarvitsisi muistaa. Tämä aiheutti kuitenkin hieman ongelmia OwnCloudin konfiguroinnin kanssa. OwnCloud on tarkoitettu käytettäväksi Apachen päällä omalla verkko osoitteellaan. nginxiä käytettäessä OwnCloud jouduttiin asentamaan käsin ja muuttamaan asetustiedostot tukemaan nginxin määrittelyjä. Owncloudin tiedostopankkina käytettiin juuri liitettyä ulkoista kovalevyä.

```
/data/ac-tekniikka.fi/www/pilviapt
```

Asennettiin tarvittavat paketit:

```
sudo apt-get install nginx openssl ssl-cert php5-cli
php5-sqlite php5-gd php5-common php5-cgi sqlite3
php-pear php-apc curl libapr1 libtool curl libcurl4-
openssl-dev php-xml-parser php5 php5-dev php5-gd
php5-fpm memcached php5-memcache varnish
```

PHP:n vaatimat muutokset määriteltiin tiedostoon:

```
sudo nano /etc/php5/fpm/php.ini
```

Laajennettiin tiedostojen maksimikokoa kymmeneen gigaan:

```
upload_max_filesize = 1000M
post_max_size = 1000M
```

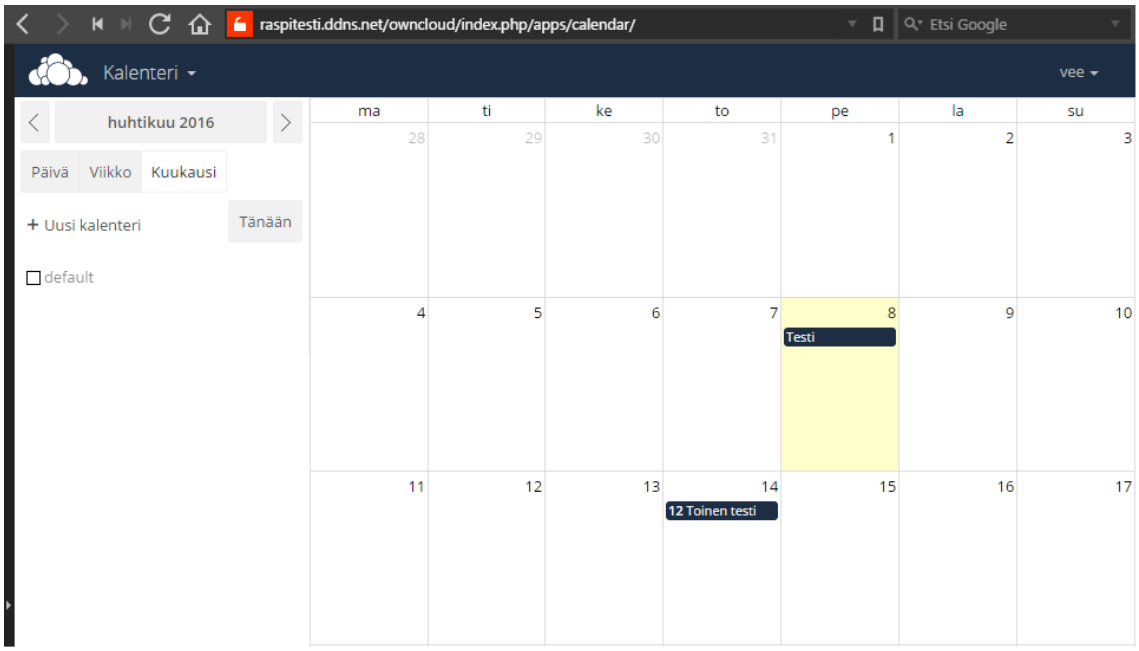
Lisättiin virtuaalisen välimuistin määrää:

```
sudo nano /etc/dphys-swapfile
```

Välimuistia on vakiona 100MB, joten kasvatamme sitä 512MB:iin:

```
CONF_SWAPSIZE=512
```

Tässä kohtaa saatiin RPi ja koko projekti niin solmuun, ettei mikään projektin osio enää toiminut. Muutaman asetustiedostoihin hukatun päivän jälkeen RPi saatiin kuitenkin taas pystyyn ja projektia päästiin jatkamaan. Kuvassa OwnCloud-kalenterisovelluksen testailua.



Kuva 19, OwnCloudin kalenterisovellus.

6.14 SSL-sertifikaattien asennus

Virallinen sertifikaatti on maksullinen, joten päädyimme allekirjoittamaan sertifikaatin itse. Ensin tehtiin sertifikaatin avaintiedosto:

```
openssl genrsa -des3 -out owncloud.key 2048
```

Sitten tehdään suojaamaton avain ja sertifikaatin tunnistetiedosto:

```
openssl rsa -in owncloud.key -out own-
cloud.key.insecure
```

```
mv owncloud.key owncloud.key.secure
mv owncloud.key.insecure owncloud.key

openssl req -new -key owncloud.key -out owncloud.csr
```

Testataan, että luotu sertifikaatti toimii:

```
openssl x509 -req -days 1825 -in owncloud.csr -
signkey owncloud.key -out owncloud.crt
```

Tehdään kansio sertifikaateille ja siirretään kaikki sertifikaattitiedostot sinne:

```
sudo mkdir /etc/nginx/certs
sudo mv owncloud.* /etc/nginx/certs/
```

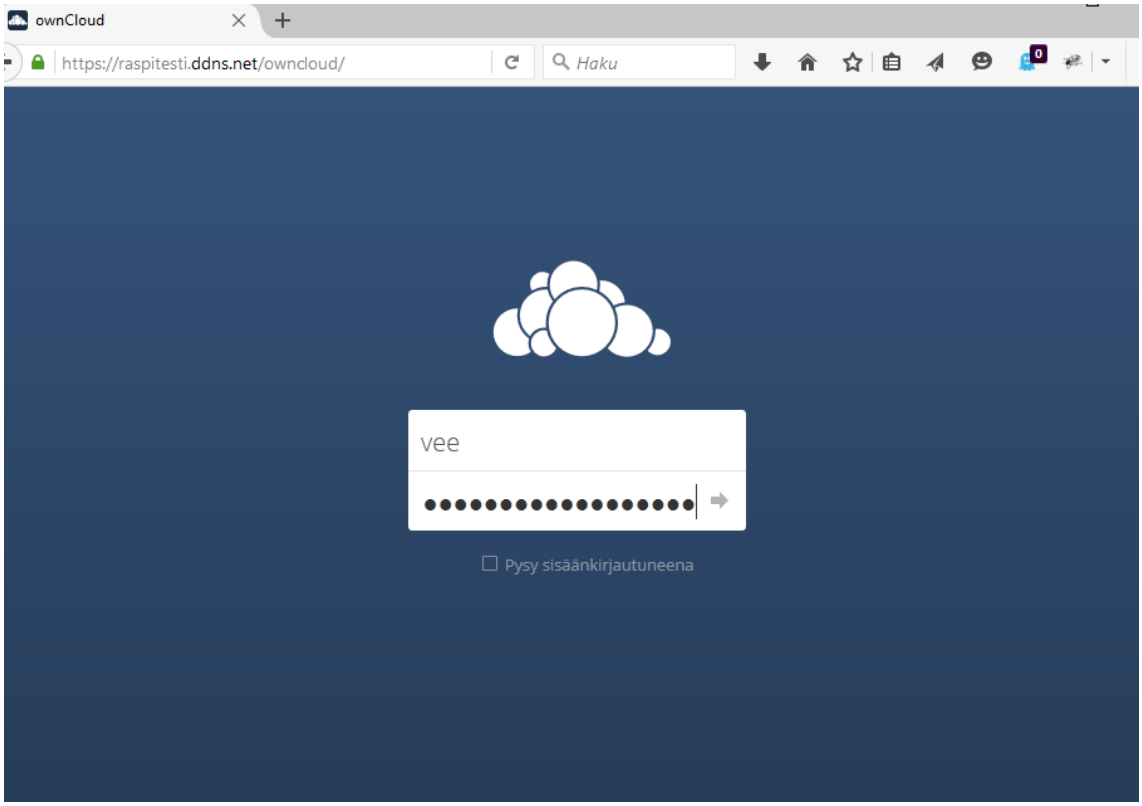
Tämän jälkeen muutetaan nginxin konfigurointia niin, että se alkaa käyttämään salattua 442 porttia salaamattoman 80 portin sijaan.

Sertifikaattien allekirjoituksen ja asetustiedostojen muokkauksen jälkeen OwnCloud saatiin toimimaan salatun HTTPS-protokollan kanssa.

Lopuksi määritellään WordPress käyttämään suojattua yhteyttä:

```
sudo nano /usr/share/nginx/www/wp-config.php

define('FORCE_SSL_ADMIN', true);
if ($_SERVER['HTTP_X_FORWARDED_PROTO'] == 'https')
       $_SERVER['HTTPS']='on';
```



Kuva 20, Vihreä lukko tarkoittaa salattua yhteyttä.

6.15 Sähköpostin asennus Postfixin, Dovecotin ja MySQL:n avulla

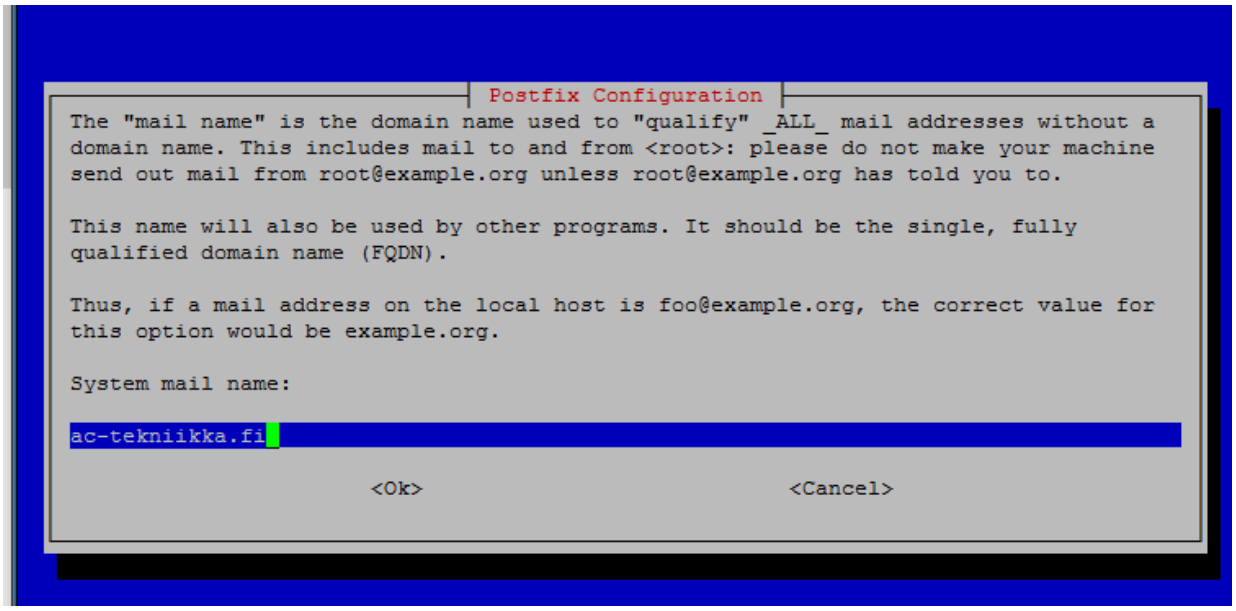
Ensin lisätään DNS-rekisteriin MX-arvo:

```
mail.ac-tekniikka.fi    MX    10    13.3.7.40
```

Sitten asennetaan RPI:lle sähköpostin vaatimat paketit:

```
sudo apt-get install postfix postfix-mysql dovecot-
core dovecot-imapd dovecot-pop3d dovecot-lmtpd dove-
cot-mysql mysql-server
```

Tämän jälkeen aukeaa Postfixin asennusvelho, joka on toteutettu terminaaliohjelmana.



Kuva 21, Postfix asetusvelho.

Asennuksen jälkeen törmättiin virheeseen:

```
Errors were encountered while processing:
dovecot-imapd
dovecot-lmtpd
dovecot-pop3d
E: Sub-process /usr/bin/dpkg returned an error code
(1)
```

Tutkimisen jälkeen selvisi, että dovecot-asennuspaketti ei osaa korvata symbolisia link-kejä uudempiin versioihin, vaan tämä on tehtävä käsin. POP3:en osalta tämä korjattiin seuraavilla komennoilla:

```
cd /etc/rc2.d
ls | grep dovecot
sudo rm -f S03dovecot
sudo apt-get remove --purge dovecot-core
```

Asennus antoi virheitä myös IPv6-määrityksistä. Määrittelyt poistettiin, koska niitä ei vielä tässä kohtaa haluttu käyttää. Sähköpostin konfigurointiin vaadittava PHPMyAdmin

asennettiin jo aiemmin, mutta sen määrittelyt täytyi vielä lisätä Nginxin konfiguraatiotiedostoon:

```
sudo nano /etc/nginx/sites-enabled/default
```

```
######  phpMyAdmin
##########################################################
    location /phpmyadmin {
        root /usr/share/;
        index index.php index.html index.htm;
        location ~ ^/phpmyadmin/(.+\.php)$ {
            root /usr/share/;
            #include fastcgi-gen.conf;
           fastcgi_pass unix:/var/run/php5-fpm.sock;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $docu-
ment_root$fastcgi_script_name;
            include /etc/nginx/fastcgi_params;
            fastcgi_buffer_size 128k;
            fastcgi_buffers 256 4k;
            fastcgi_busy_buffers_size 256k;
            fastcgi_temp_file_write_size 256k;
            fastcgi_read_timeout 240;
        }
        location ~*
^/phpmyadmin/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
            root /usr/share/;
        }
    }
    location /phpMyAdmin {
        rewrite ^/* /phpmyadmin last;
    }
```

Tämän jälkeen PHPMyAdmin on hallittavissa osoitteessa ac-tekniikka.fi/php.

Sitten konfiguroitiin MySQL-tietokanta, joka sisältää kolme taulua: yksi verkkotunnukselle, toinen sähköpostiosoitteille ja kolmas kryptatuille salasanoille.

Ensin tehtiin tietokanta:

```
mysqladmin -p create mailserver
```

Asennuksen helpottamiseksi annettiin SSH-käyttäjälle täydet oikeudet mailserver-tietokantaan. Kirjaudutaan mailserver-tietokantaan komennolla:

```
mysql -p mailserver
```

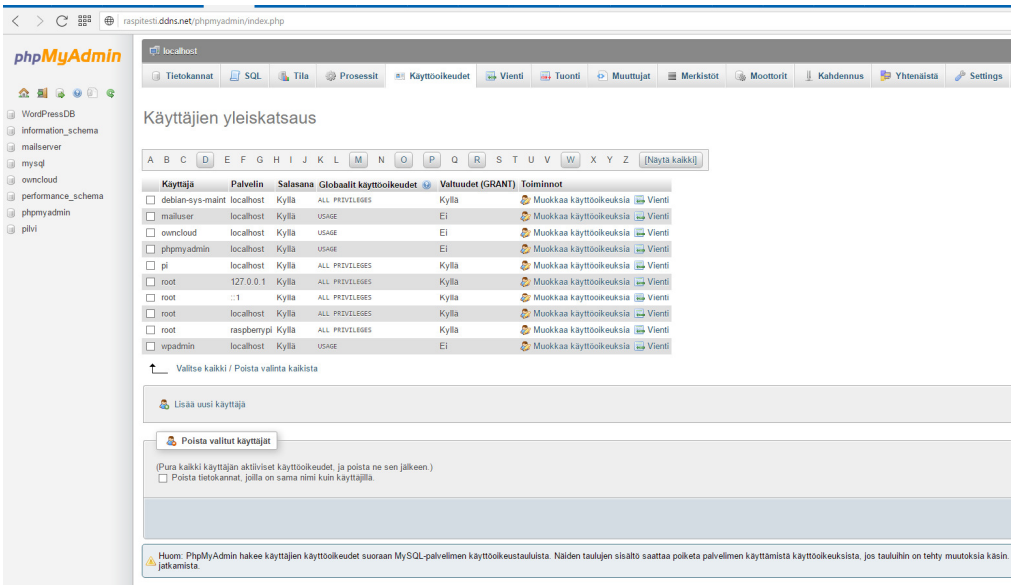
Tehdään uusi MySQL-käyttäjä kyseiselle tietokannalle:

```
CREATE USER 'mailuser'@'localhost' IDENTIFIED BY
`tamaonkosalasanasi;
```

Ja annetaan sille täydet oikeudet tietokantaan:

```
GRANT ALL PRIVILEGES ON mailserver.* TO 'mailu-
ser'@'localhost';
```

Oikeuksia voidaan katsoa helpoiten phpMyAdmin -hallintasivustolta:



Kuva 22, phpMyAdminin hallinta.

Seuraavaksi tehtiin taulu nimeltä virtual_domains ja lisättiin siihen kaksi solua: toinen ID:lle ja toinen verkkotunnukselle.

```
CREATE TABLE `virtual_domains` (`id` int(11) NOT
NULL auto_increment, `name` varchar(50) NOT NULL,
PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHAR-
SET=utf8;
```

Sitten tehtiin taulu virtual users, johon lisättiin solut ID:lle, salasanalle, ja sähköposti-osoitteelle.

Teimme vielä sähköpostin uudelleenlähetyksen mahdollistavan ”alias”-taulun

```
CREATE TABLE `virtual_aliases` (`id` int(11) NOT
NULL auto_increment,`domain_id` int(11) NOT
NULL,`source` varchar(100) NOT NULL,`destination`
varchar(100) NOT NULL,PRIMARY KEY (`id`),FOREIGN KEY
(domain_id) REFERENCES virtual_domains(id) ON DELETE
CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tietokannan luonnin jälkeen sinne lisättiin olemassa olevien osoitteiden lista

```
INSERT INTO mailserver.virtual_domains (id ,name)
VALUES
(1, ‘raspitesti.ddns.net’),
(2, ’maili.raspitesti.ddns.net’),
(3, ’mail.raspitesti.ddns.net’),
(4, ‘localhost.raspitesti.ddns.net’),
(5, ‘wdraspi.raspitesti.ddns.net’);
```

Seuraavaksi luotiin sähköpostiosoitteet ja vakiosalasanat käyttäjille. Tässä yhteydessä voidaan kryptata salasanat automaattisesti komennolla:

```
INSERT INTO mailserver.virtual_users (id, domain_id,
password , email) VALUES (`1`, `1`,
ENCRYPT(`Salasana1`, CONCAT(`$6$`, SUB-
STRING(SHA(RAND()), -16))), `esimerkki@ac-
tekniikka.fi`);
```

Samaan tyyliin lisättiin myös sähköpostin aliakset.

Postfix-konfigurointi tehtiin omaan asetustiedostoonsa:

```
nano /etc/postfix/main.cf
```

MySQL-tietokanta osoitettiin postfixille lisäämällä mysql-virtual-mailbox-domains.cf -tiedostoon käyttäjätiedot ja tietokannan nimi:

```
user = meikamandoliini
    password = tosipitkäsalasana11eiriita
    hosts = 127.0.0.1
    dbname = mailserver
query = SELECT 1 FROM virtual_domains WHERE
name='%s'
```

Testataan, löytyykö tehty mailiosoite tietokannasta:

```
service postfix restart

postmap -q testi@raspitesti.ddns.fi
mysql:/etc/postfix/mysql-virtual-mailbox-maps.cf
```

Dovecotin asetustiedossa otettiin käyttöön submission ja smtps, jotta saatiin sähköposti ohjattua salattuna porttien 587 ja 465 lisäksi myös porttiin 25.

```
sudo nano /etc/dovecot/conf.d/10-master.conf
```

```
==========================================================================
====
# service type  private unpriv  chroot  wakeup  maxproc command + args
#               (yes)   (yes)   (yes)   (never) (100)
#
==========================================================================
====
smtp       inet  n       -       -       -       -       smtpd
#smtp      inet  n       -       -       -       1       postscreen
#smtpd     pass  -       -       -       -       -       smtpd
#dnsblog   unix  -       -       -       -       0       dnsblog
#tlsproxy  unix  -       -       -       -       0       tlsproxy
submission inet  n       -       -       -       -       smtpd
#  -o syslog_name=postfix/submission
#  -o smtpd_tls_security_level=encrypt
#  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
#  -o milter_macro_daemon_name=ORIGINATING
smtps     inet   n       -       -       -       -       smtpd
#  -o syslog_name=postfix/smtps
```

```
#  -o smtpd_tls_wrappermode=yes
#  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_client_restrictions=permit_sasl_authenticated,reject
#  -o milter_macro_daemon_name=ORIGINATING
```

Dovecotin yleiset asetukset löytyivät tiedostosta:

```
nano /etc/dovecot/dovecot.conf
```

Otettiin IPv6:nen pois käytöstä ja lisättiin muutama rivi asetustiedoston loppuun, jotta saatiin IMAP-sähköpostit toimintaan:

```
namespace inbox {
inbox = yes
}
```

Sitten seuraava asetustiedosto:

```
nano /etc/dovecot/conf.d/10-mail.conf
```

Tiedostoon lisättiin sähköpostien säilytyspaikka ja ryhmän tieto:

```
mail_location = maildir:/data/mail/vhosts/%d/%n
mail_privileged_group = mail
```

Seuraavaksi tehtiin kansio sähköposteille:

```
mkdir /data/mail
ls -ld /data/mail
mkdir -p /data/mail/vhosts/ac-tekniikka.fi
```

Luotiin sähköpostin hallintaan käyttäjä ja säädetään oikeudet:

```
groupadd -g 5000 vmail
useradd -g vmail -u 5000 vmail -d /data/mail
chown -R vmail:vmail /data/mail
```

Lisätään osoitus käyttäjätietokantaan:

```
nano /etc/dovecot/conf.d/auth-sql.conf.ext

userdb {
  driver = static
  args = uid=vmail gid=vmail
home=/data/mail/vhosts/%d/%n
}
```

Liitetään dovecot SQL tietokantaan käyttäjänimellä.

Muutetaan liitostiedot:

```
nano /etc/dovecot/dovecot-sql.conf.ext

connect = host=127.0.0.1 dbname=mailserver us-
er=mailuser password=salasanasi
default_pass_scheme = SHA512-CRYPT
password_query = SELECT email as user, password FROM
virtual_users WHERE email='%u';
```

Lopuksi määritellään vmail käyttäjä Dovecotin omistajaksi:

```
chown -R vmail:dovecot /etc/dovecot
chmod -R o-rwx /etc/dovecot
```

7 Yhteenveto

Työn tavoitteena oli tehdä asiakkaalle toimiva IaaS-pilvipalveluympäristö mahdollisimman kustannustehokkaasti. Työssä huomioitiin alkuinvestointi ja käyttöönotetun ympäristön jatkuvat kulut. Lisäksi tavoitteena oli tutustua pilvipalveluihin, minitietokoneisiin, Linuxiin, julkaisualustoihin ja näiden tietoturvaan.

Tekijällä ei ollut juurikaan aikaisempaa kokemusta työssä käytetyistä tekniikoista tai alustoista. Projektin edetessä törmättiin yhteensopivuusongelmiin käytetyn web serverin (nginx) ja yksityisen tallennuspalvelun kanssa (OwnCloud), jotka eivät olleet keskenään yhteensopivia. Yhteensopivuusongelmat ratkaistiin asentamalla OwnCloud käsin ja määrittelemällä nginx:in asetukset uudelleen.

Asiakkaan tavoite mahdollisimman kustannustehokkaasta ratkaisusta aiheutti suurimmat ongelmat työn tekijälle. Kustannustehokkuuden nimissä käytettiin ilmaisia työkaluja, joiden kanssa usein tulee vastaan niiden rajoittuneet ominaisuudet. Tästä esimerkkinä itse allekirjoitettuun SSL-sertifikaattiin liittyvät ongelmat ja nimipalvelun rajoittuneet hallintamahdollisuudet. Haasteita aiheutti myös asiakkaan aikataulun sovittaminen insinöörityön tekijän aikataulujen kanssa, asiakkaan asettamat tavoitteet toteutettiin kuitenkin projektille asetetussa aikataulussa.

Palvelu saatiin valmiiksi ja luovutettiin asiakkaan testiin. Palvelu rajattiin testivaiheessa niin, ettei ulkopuolisilla ollut mahdollisuutta nähdä esimerkiksi yrityksen uudistettua nettisivua tai muuta sisältöä. Palvelun tietoturva arvioitiin uudelleen insinöörityön tekijän ehdotuksesta ja laajennettiin tukemaan salattuja yhteyksiä tallennuspalvelun turvaamiseksi.

Toteutustapa oli mielestäni onnistunut ja sopii pienille yrityksille ja organisaatioille laitteistoa hieman päivittämällä. Asiakas oli kustannustehokkaaseen toteutukseen tyytyväinen, palvelu otetaan käyttöön uudistetulla raudalla, tietoturvaa entisestään parantavaa TLS-sertifikaattia ja nimipalvelua hyödyntäen.

Lähteet

1 Pilvipalveluiden palvelumallit. Verkkodokumentti. Saatavissa: https://aws.amazon.com/types-of-cloud-computing/ Luettu 03.02.2016.

2 What is cloud computing. Verkkodokumentti. Saatavissa: http://www.interoute.com/cloud-article/what-cloud-computing Luettu 03.02.2016.

3 Petteri Heino, Pilvi palvelut – Cloud computing. Kariston Kirjapaino Oy 2010 Hämeenlinna.

4 ODROID X4 arvostelu. Verkkodokumentti. Saatavissa: http://www.mikronauts.com/hardkernel/hardkernel-odroid-xu4-review/4/ Luettu 16.02.2016.

5 Virrankulutukset. Verkkodokumentti. Saatavissa: http://www.cnx-software.com/wp-content/uploads/2015/02/Raspberry_PI_ODroid_Banana_Power_Consumption1.png Luettu 03.02.2016.

6 What is a Raspberry Pi. Verkkodokumentti. Saatavissa: https://www.raspberrypi.org/help/what-is-a-raspberry-pi/ Luettu 03.03.2016.

7 Raspberry Pi:n komponenttisijoittelu. Verkkodokumentti. Saatavissa: http://www.cnet.com/products/raspberry-pi-model-b-plus/ http://cnet2.cbsistatic.com/hub/i/r/2014/07/14/c2133bb2-c63d-4cc2-baa6-c5349452c6a2/resize/770x578/18e8be6c71c24b9b3d83d1dbe905d3a1/raspberry-pi-b-plus3info.jpg Luettu 03.03.2016.

8 SD Association. Verkkodokumentti. Saatavissa: https://www.sdcard.org Luettu 03.03.2016.

9 WordPress-asennus. Verkkodokumentti. Saatavissa: https://codex.wordpress.org/Installing_WordPress Luettu 10.03.2016.

10 FastCGI-asennus. Verkkodokumentti. Saatavissa: https://www.digitalocean.com/community/tutorials/how-to-setup-fastcgi-caching-with-nginx-on-your-vps Luettu 10.03.2016.

11 Nginx-asetukset ja vianetsintä. Verkkodokumentti. Saatavissa: https://www.nginx.com/resources/wiki/ Luettu 07.04.2016.

12 LAMP-serverin asennus. Verkkodokumentti. Saatavissa: https://www.pestmeester.nl/index.html#8.1 Luettu 07.04.2016.

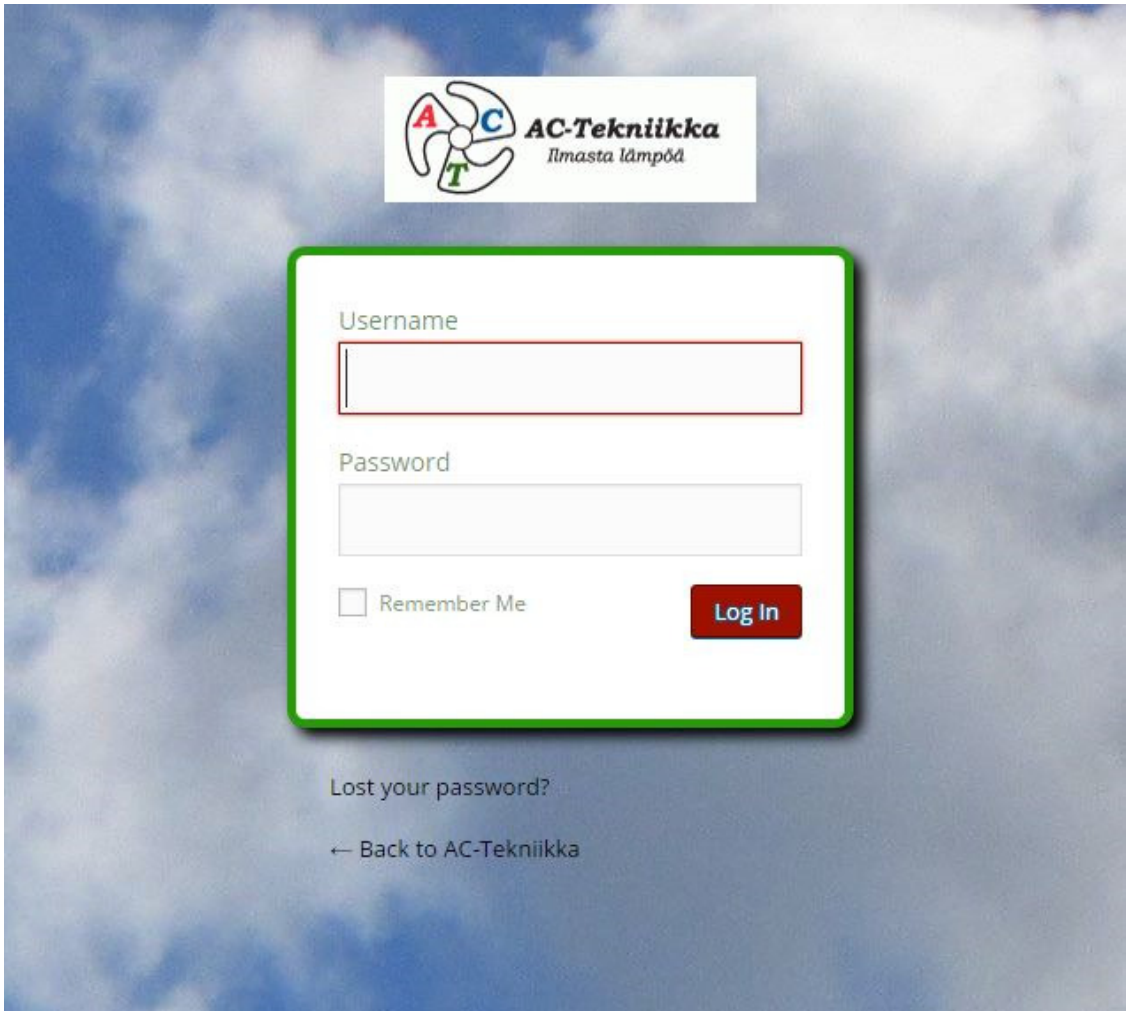
13 Sähköpostin konfigurointi. Verkkodokumentti. Saatavissa: https://www.linode.com/docs/email/postfix/email-with-postfix-dovecot-and-mysql Luettu 14.04.2016.

Liitteet

WordPress sivuston käyttöohjeet

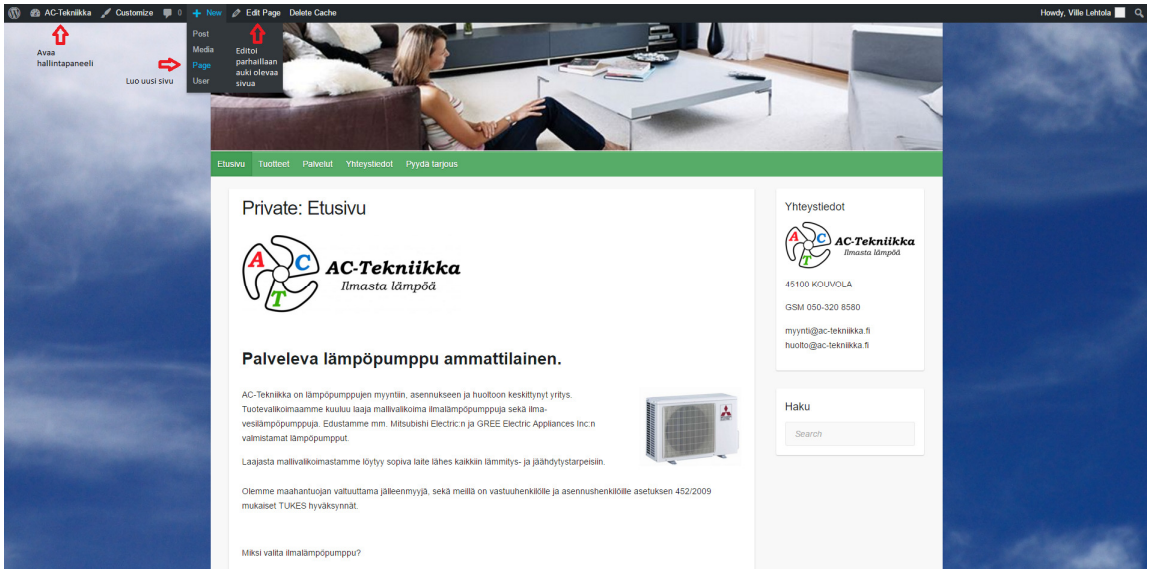
7.1 Aloitus

Käynnistä selain ja surffaa osoitteeseen
www.ac-tekniikka.fi/hallinta



Käyttäjänimesi on lähetetty sähköpostiisi ja salasanasi on toimitettu tekstiviestillä.
Salasanan resetointipyynnöt osoitteeseen:

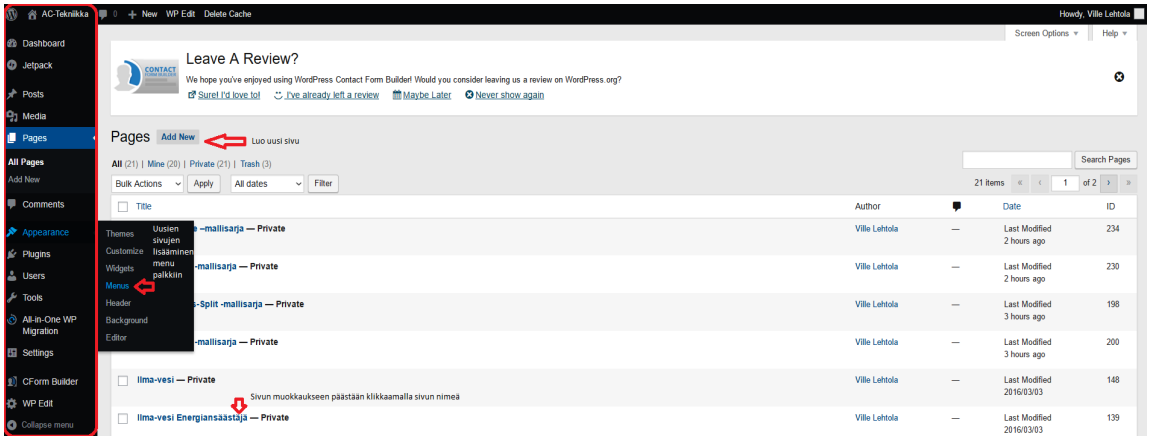
Kirjautumisen jälkeen avautuu etusivu WordPressin työkaluilla. (Kuvat saat isommiksi vanhoilla Office versioilla: Ctrl + hiiren rulla ja O360 klikkaamalla kuvaa)



Kuvaan merkitty nuolilla eniten käytetyt pikakomennot.

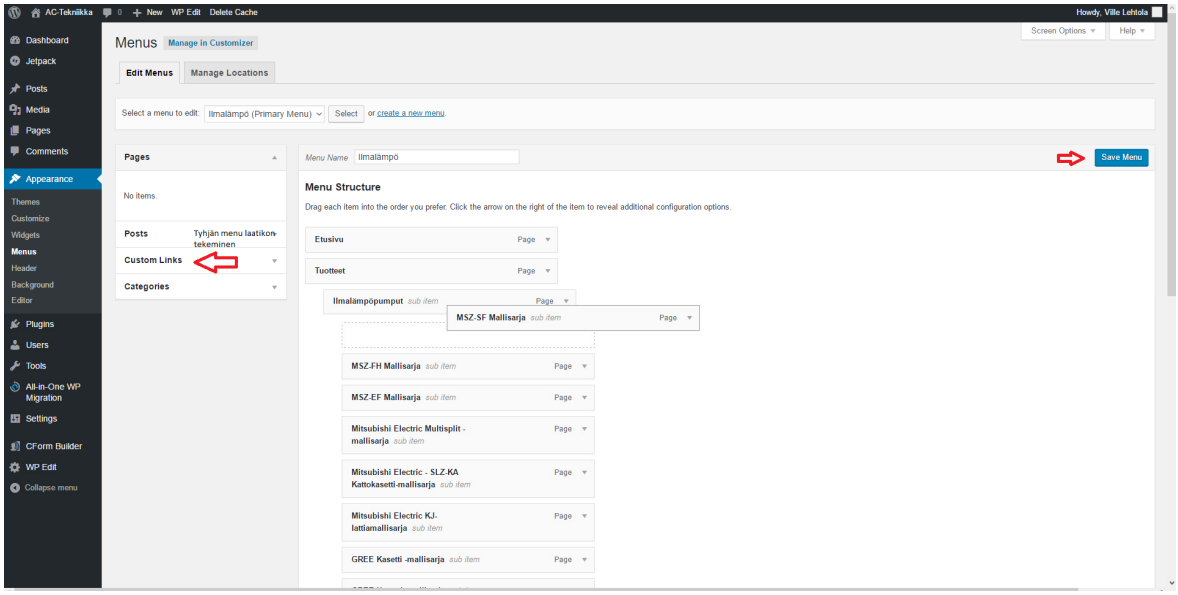
7.2 Sivupalkki ja navigointi

Mökin kuvaa painamalla päästään laajempiin asetuksiin. Vasemmasta valikosta löytyvät kaikki sivuston ylläpitoon vaadittavat osiot.



7.3 Valikon hallinta

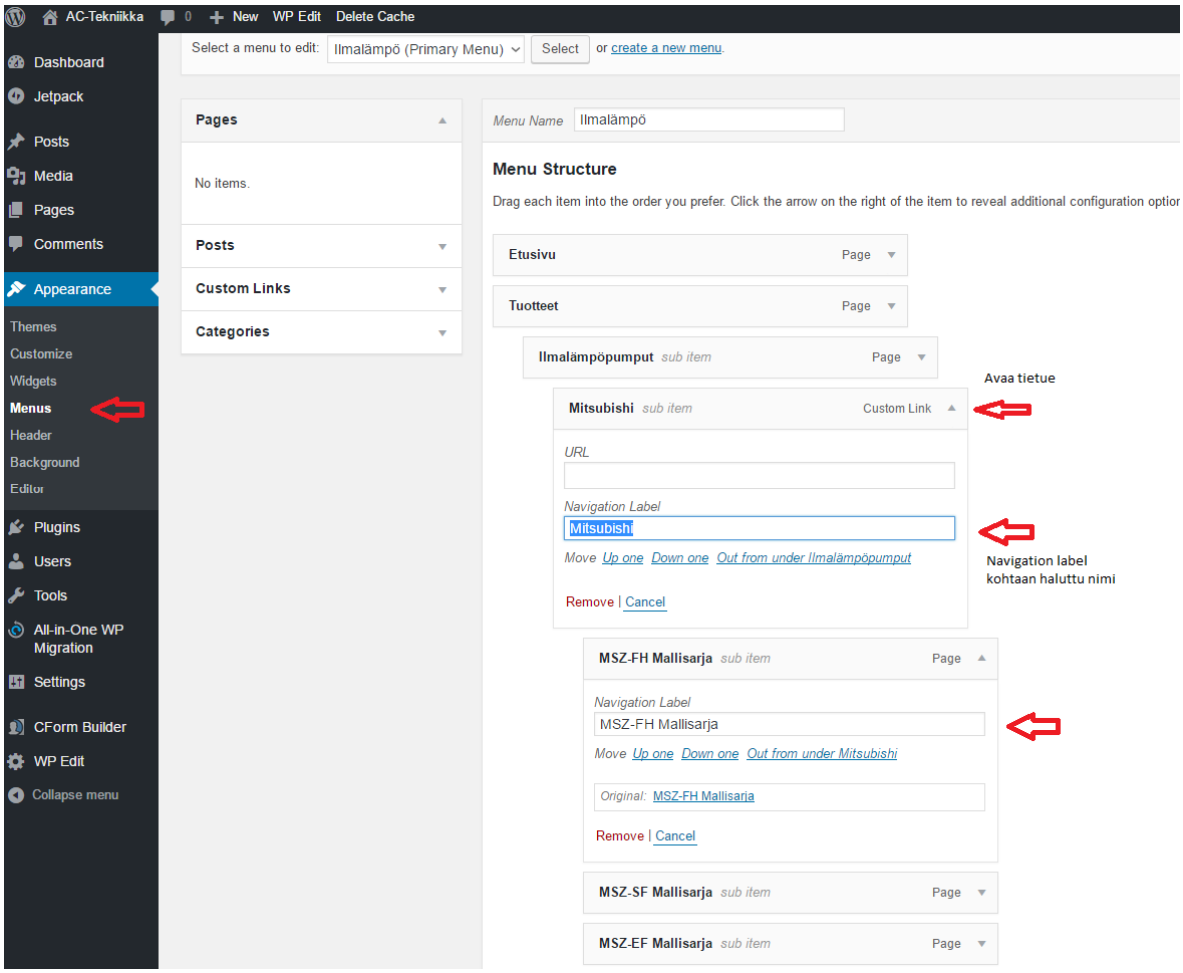
Uudet sivut lisäytyvät automaattisesti valikkoon ja niiden paikkoja voi muuttaa drag n drop periaatteella. Pääsivut ovat vasemmalla ja alasivut oikealla.



Tyhjän valikkolaatikon luonti onnistuu klikkaamalla custom link.

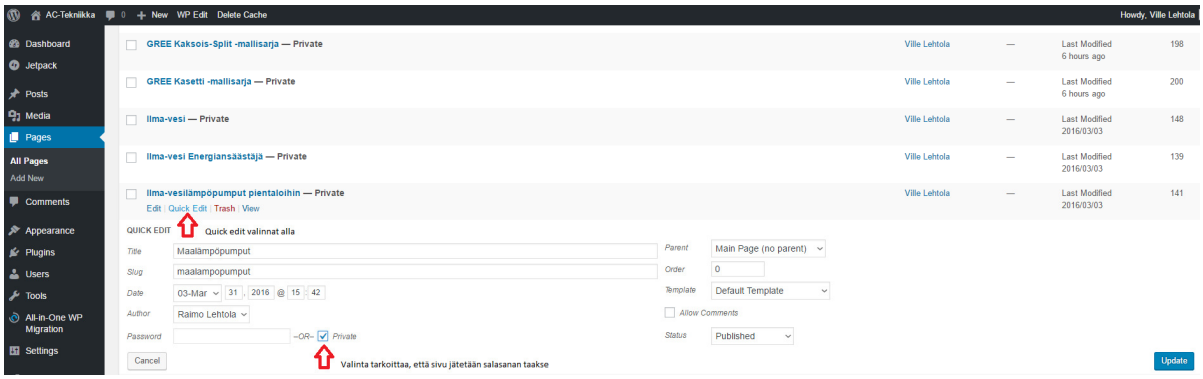
URL kenttään: javascript: void(0);

Link Text kenttään nimi jonka halutaan näkyvän valikossa.



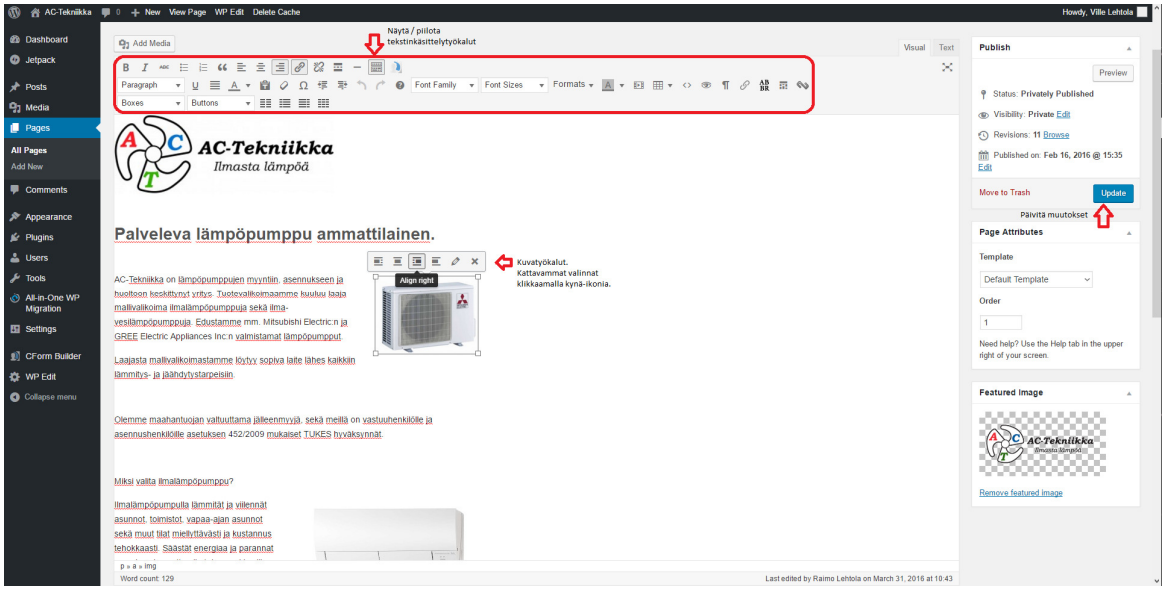
Custom link tarkoittaa pelkkää valikkolinkkiä ja page tarkoittaa normaalia verkkosivua.

7.4 Sivun piilottaminen valikkoon lisäämisen jälkeen



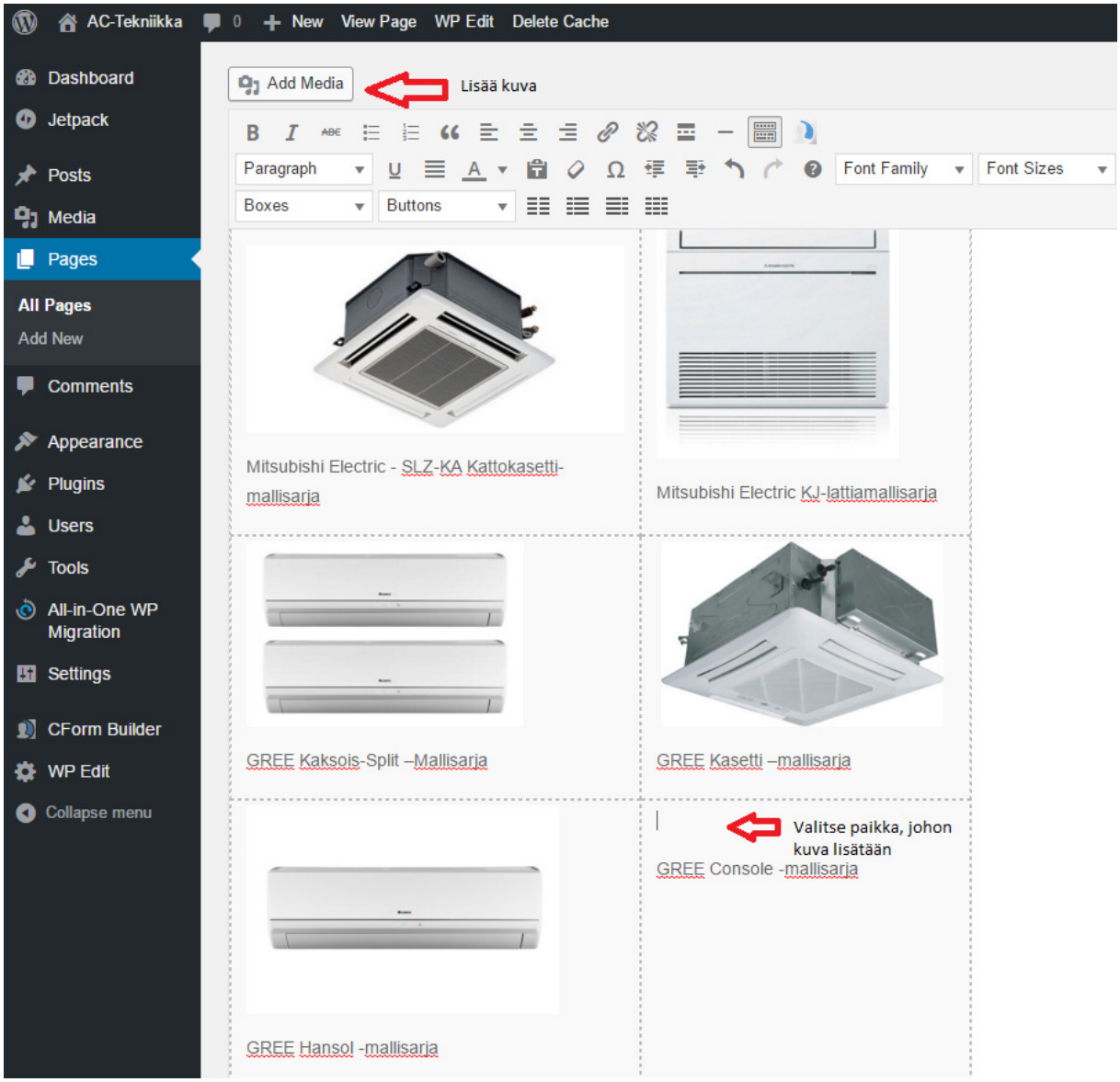
Viedään hiiri sivun nimen päälle, jolloin ilmestyvät kuvassa näkyvät valinnat. Valitse Quick Edit ja lisää Private ruutuun ruksi. Tämän jälkeen sivun nimen jälkeen näkyy maininta –Private.

7.5 Sivun muokkaus

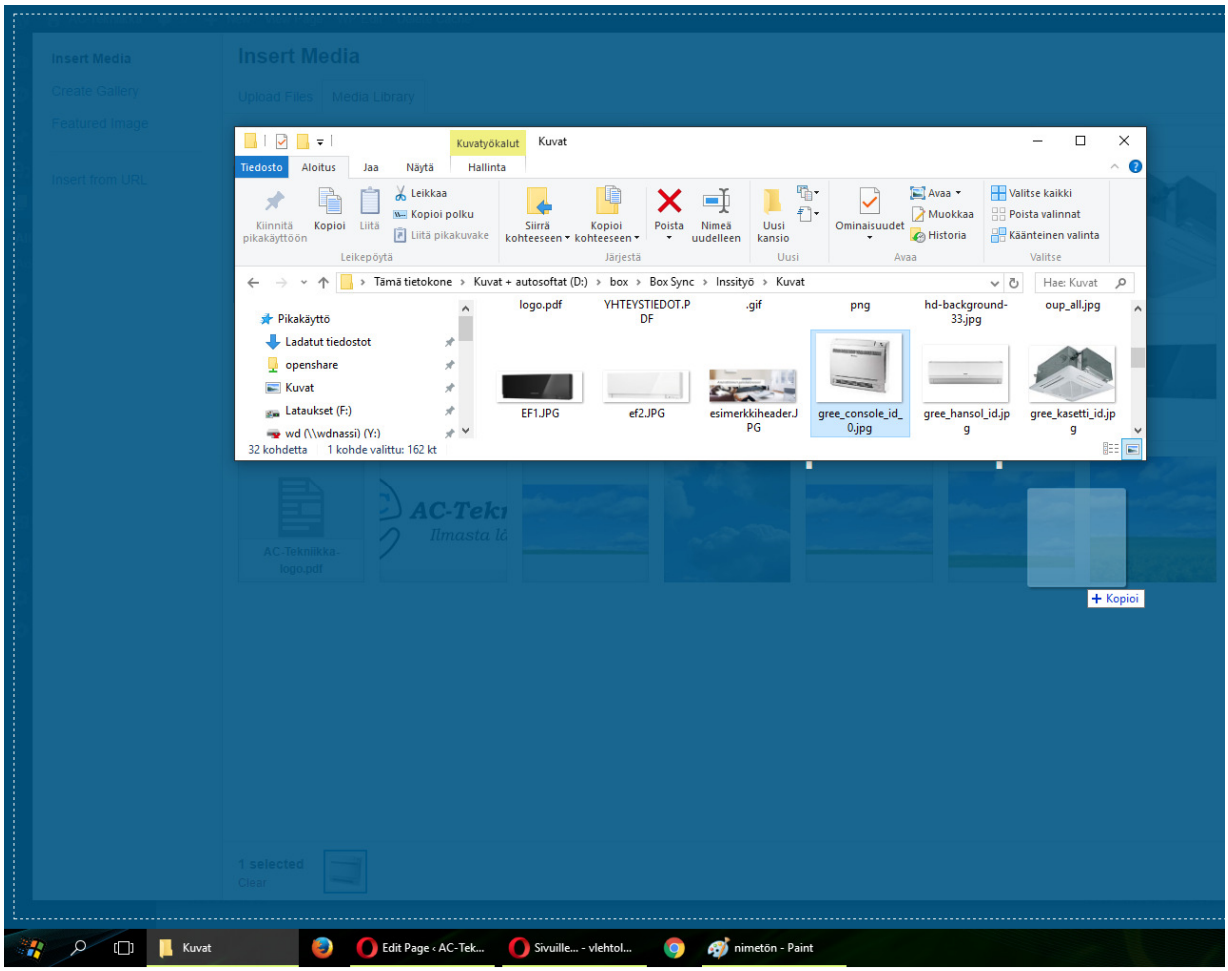


Punaisen nelikulmion sisältä löytyvät käytössä olevat työkalut. Tekstin ja kuvien lisääminen onnistuu myös kopioimalla teksti tai kuva muusta sijainnista ja liittämällä tähän editoriin.

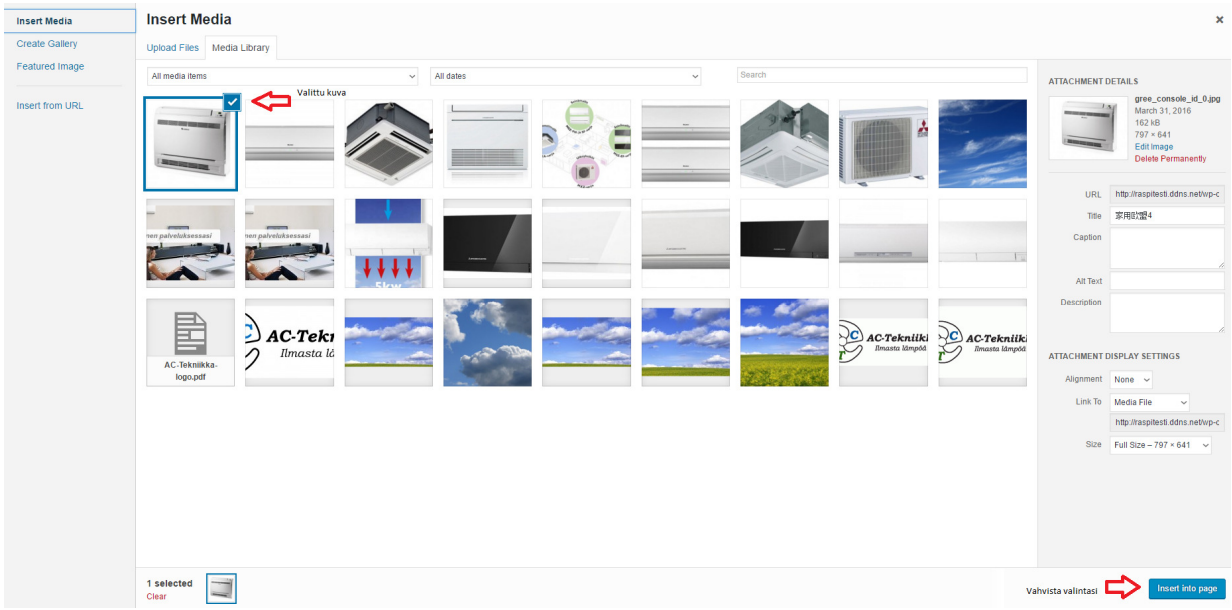
7.6 Kuvan lisääminen ja sen linkittäminen



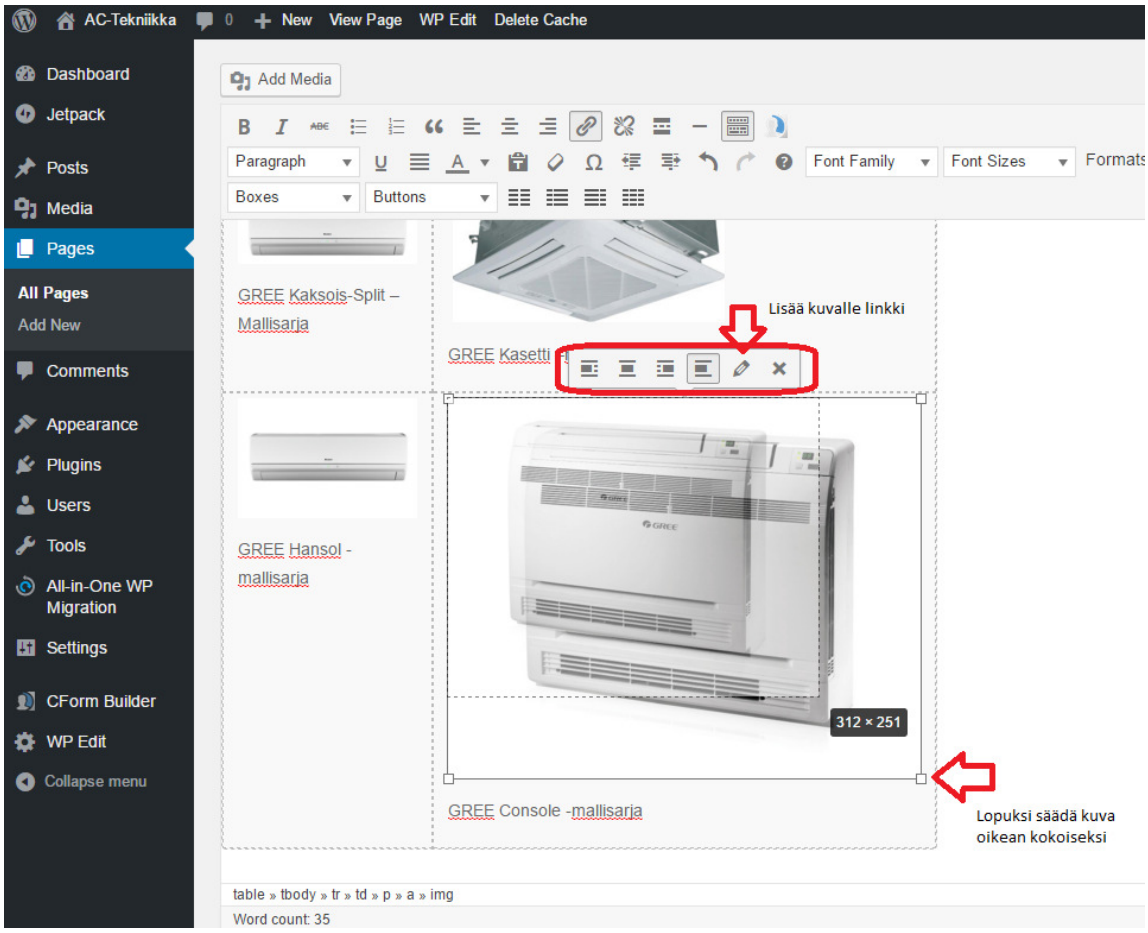
Kuvan lisääminen tietokoneeltasi onnistuu helpoiten drag n drop periaatteella.



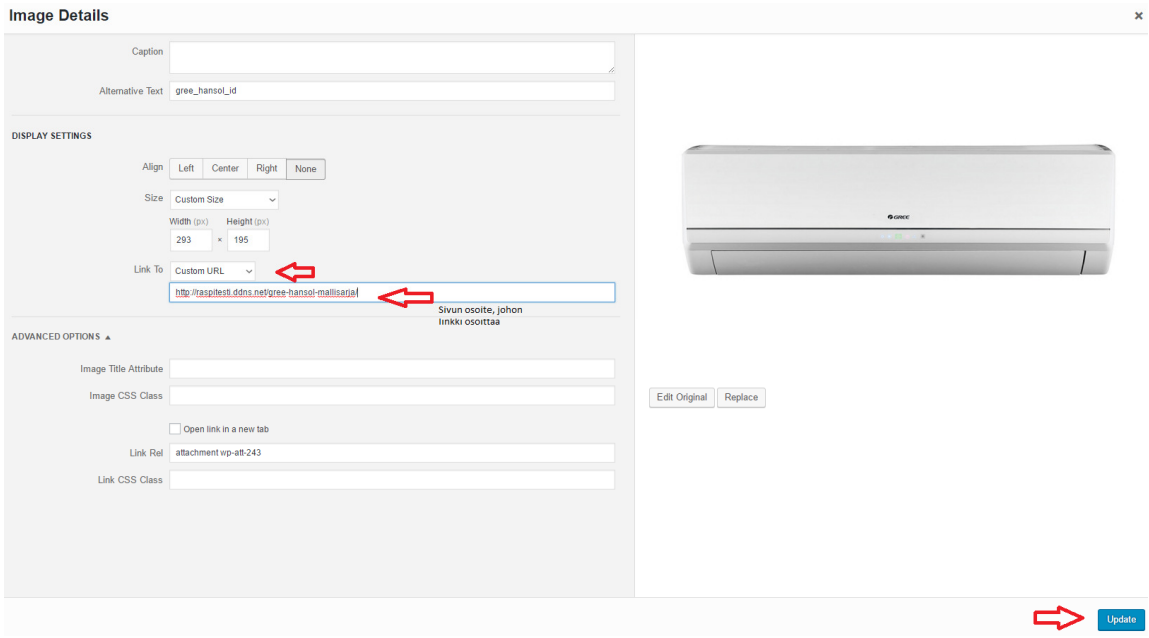
Lataamisen jälkeen kuva ilmestyy galleriaan valmiiksi valittuna. Kuva lisätään sivulle Insert into page napilla.



Kuva on yleensä liian suuri, joten sitä joutuu pienentämään kuvan ohjeen mukaan



Kynä-ikonia painamalla saadaan kuvan lisäasetukset näkyviin. Lisäasetuksista löytyy mm. kuvan muuttaminen linkiksi.



7.7 Liitännäisten päivittäminen

Klikkaa sivupalkista updates, select all ja update plugins. Teemat päivitetään samalta sivulta.