



**SAVONIA**

■ OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO  
TEKNIIKAN JA LIIKENTEEN ALA

# PALVELINPOHJAINEN PILVITALLENNUSPALVELU

Server-based cloud storage service

TEKIJÄ/T: Toni Viertomanner

Koulutusala Tekniikan ja liikenteen ala	
Koulutusohjelma Tietotekniikan koulutusohjelma	
Työn tekijä(t) Toni Viertomanner	
Työn nimi Palvelinperusteinen pilvitalennuspalvelu	
Päiväys 31.5.2017	Sivumäärä/Liitteet 19
Ohjaaja(t) Lehtori Pekka Granroth, lehtori Veijo Pitkänen	
Toimeksiantaja/Yhteistyökumppani(t) Kareta Oy	
Tiivistelmä <p>Tämän opinnäytetyön aiheena oli tehdä yritykselle omalla laitekoonpanolla toimiva pilvitalennuspalvelu, koska useat valmiit ratkaisut tarjoavat rajoitetun tallennustilan tai muuten suppeat ominaisuudet.</p> <p>Työn tekeminen alkoi ohjelmiston, käyttöjärjestelmän sekä palvelinkoneen komponenttien valinnalla. Valintoihin vaikuttivat tarkoituksenmukaisuus, projektin budjetti ja ylläpito henkilöstön osaaminen. Seuraavaksi suunniteltiin kokonaisuus, jossa kiinnitettiin erityisesti huomiota tietoturvaan ja käytettävyyteen loppukäyttäjän kannalta. Sitten projekti toteutettiin palvelinkoneen kasaamisesta ja käyttöjärjestelmän sekä ohjelmiston asentamisesta kohti pienempiä, mutta oleellisia yksityiskohtia. Lopuksi suoritettiin testaus, asennettiin päätteille synkronointisovellus sekä mietittiin mahdollisia jatkokehitysideoita palvelimelle.</p>	
Avainsanat ownCloud, Linux, Ubuntu, palvelin, pilvitalennus	

Field of Study Technology, Communication and Transport			
Degree Programme Degree Programme in Information Technology			
Author(s) Toni Viertomanner			
Title of Thesis Server-based cloud storage service			
Date	31.5.2017	Pages/Appendices	19
Supervisor(s) Lecturer Pekka Granroth, Lecturer Veijo Pitkänen			
Client Organisation /Partners Kareta Oy			
<p>Abstract</p> <p>The subject of this thesis was to create cloud storage service working on own device composition for a company, because several existing solutions are offering limited storage space or other restricted features.</p> <p>The first step of the work was choosing software, operating system and components of the server computer. Aspects that affected to decisions were their suitability to the purpose, a budget of the project and knowledge of an administrative staff. Then each part of the project was outlined which especially focused on security and usability of the end user's view. After that, the project was carried out from building the server computer, installing the operating system and software all the way to the smaller but essential details. In the end, testing was performed, the synchronization application was installed to client computers and possible future improvements were considered.</p>			
Keywords ownCloud, Linux, Ubuntu, server, cloud storage			

## ESIPUHE

Haluan kiittää toimeksiantajayritys Kareta Oy:ta mahdollisuudesta tämän mielenkiintoisen projektin toteutukseen, sekä Jouni Nousiaista ulkopuolisesta konsultaatiosta. Kiitos myös lehtori Pekka Granrothille ja lehtori Veijo Pitkäselle työn ohjauksesta ja laadukkaasta opetuksesta.

Kuopiossa 31.05.2017

Toni Viertomanner

## SISÄLTÖ

LYHENTEET JA MÄÄRITELMÄT .....	6
1 JOHDANTO .....	7
2 SUUNNITTELU .....	8
2.1 Palvelinkoneen komponentit .....	8
2.2 Käyttöjärjestelmä .....	8
2.3 Ohjelmistosovellus ownCloud .....	8
2.4 Tietoturva .....	9
3 TOTEUTUS.....	10
3.1 Palvelinkoneen kokoaminen ja Linux Ubuntu 14.04 LTS Server-käyttöjärjestelmän asennus.....	10
3.2 Kovalevyjen peilaus RAID1-tekniikalla .....	11
3.3 Käyttöjärjestelmän valmistelu tiedostonjakopalvelua varten .....	12
3.4 OwnCloud-tiedostonjakosovelluksen asennus.....	12
3.5 Tietoturva .....	15
4 TESTAUS .....	18
5 JATKOKEHITYS .....	18
6 YHTEENVETO.....	19
LÄHTEET .....	19

## LYHENTEET JA MÄÄRITELMÄT

Komponentti = Tietokoneen fyysinen rakenneos

Linux Ubuntu = Avoimeen lähdekoodiin perustuva käyttöjärjestelmä

ownCloud = Avoimeen lähdekoodiin perustuva tiedostonjakopalvelu-ohjelmisto

TLS tai SSL = Salausprotokolla tietoliikenteen suojaamiseen IP-verkkojen yli

HTTP-protokolla = Hypertekstin siirtoprotokolla tiedonsiirtoon WWW-palvelimen ja selaimen välillä

HTTPS = TLS/SSL-protokollan ja HTTP-protokollan yhdistelmä

RAID-tekniikka = Tekniikka, jolla tietokoneen vikasietoisuutta ja/tai nopeutta kasvatetaan yhdistämällä kaksi tai useampi kovalevy yhdeksi loogiseksi levyksi

Hostname = Yksilöllinen nimi, jolla muut verkon käyttäjät tunnistavat palvelimen

SWAP-osio = Keskusmuistin muistitilan loppuessa kovalevylle siirrettävälle datalle tarkoitettu osio kovalevyllä

Root = Unix/Linux-järjestelmien pääkäyttäjä

Apache2 = Ilmainen avoimeen lähdekoodiin perustuva web-palvelinohjelmisto

## 1 JOHDANTO

Pilvitallennuspalvelut ovat nykyisin hyvin yleisiä tiedostojen säilytysratkaisuja, koska esimerkiksi laitteen rikkoutuessa tiedostot ovat verkkoyhteyden päässä palvelimella turvassa ja ne voidaan avata millä tahansa verkkoon kytketyllä laitteella. Toinen tärkeä etu pilvitallennuksessa on tiedostojen jatkamahdollisuus laitteiden, käyttäjien tai esimerkiksi työyhteisön välillä.

Tässä opinnäytetyössä käsittelen yksityiskohtaisesti oman pilvitallennusratkaisun käyttöönoton yritykselle, jolla ei aikaisemmin ollut omaa palvelinta. Vastaava ratkaisu soveltuu myös hyvin kotikäyttöön, mutta laitteiston osalta kannattaa ottaa huomioon riittävä jäähdytys, jos järjestelmän on tarkoitus olla päällä jatkuvasti.

## 2 SUUNNITTELU

Suunnittelussa tuli ottaa huomioon projektin budjetti, tilankäyttö, riittävä tietoturva sekä loppukäyttäjää ajatellen helppokäyttöisyys ja synkronoitavuus.

### 2.1 Palvelinkoneen komponentit

Komponenttien valinnassa tuli erityisesti ottaa huomioon prosessorin, emolevyn ja kahden kovalevyn laatu sekä RAM-muistin riittävyys ja kotelon riittävä koko ilmanvaihtoa varten.

Lista palvelinkoneeseen valituista komponenteista:

- Intel Haswell Core i3-4150 prosessori
- AsRock H97M Anniversary emolevy
- Kingston 8GB HyperX Fury muisti
- Western Digital 500GB WD Blue kovalevyjä 2 kappaletta
- Silverstone 300W Strider, 80 Plus Bronze virtalähde
- Thermaltake Core V21 ikkunallinen Micro-ATX-kotelo

### 2.2 Käyttöjärjestelmä

Käyttöjärjestelmäksi valikoitui Linux Ubuntu 14.04 LTS Server kustannussyistä sekä sen riittävän varmuuden, monipuolisuuden, helppokäyttöisyyden ja pitkän tuen takia. Ainoa vartenotettava toinen vaihtoehto oli Linux-pohjainen Fedora Server, mutta se koitui projektin tarkoitukseen liian haastavaksi ylläpitohenkilöstön osalta.

### 2.3 Ohjelmistosovellus ownCloud

OwnCloud tiedostonjakosovellus on ilmainen Linux-tyyppisiin käyttöjärjestelmiin tarkoitettu valmis pilvitallennusohjelmistokokonaisuus, joka on ohjeita noudattaen suhteellisen yksinkertainen asentaa. Sen pystyy myös asentamaan suoraan isäntäpalvelimelle. Projektiin ownCloud oli riittävän monipuolinen, koska ylläpitäjä pystyy lisäämään ja muokkaamaan käyttäjätunnuksia, sekä rajoittamaan näiden oikeuksia. Lisäksi tiedoston alkuperäinen lisääjä pystyy rajaamaan käyttäjäryhmän tai tietyt käyttäjät, joille tiedosto näkyy. Myös luku- ja kirjoitusoikeudet ovat määriteltävissä.

OwnCloudiin on web-käyttöliittymän lisäksi tehty myös synkronointisovellukset Windows- ja Linux-käyttöjärjestelmiin sekä yleisimpiin mobiilikäyttöjärjestelmiin, kuten Android:lle ja iOS:lle. Tämän vuoksi asennusvaiheen jälkeen käyttö on loppukäyttäjälle yhtä helppoa kuin yleisimpien pilvitallennuspalveluiden käyttö.



## 2.4 Tietoturva

Tietoturvan suunnittelussa tärkeimmiksi asioiksi nousivat varmuuskopiointi, virustorjunta, palomuri ja tiedonsiirron suojaus HTTPS-protokollalla. Varmuuskopiointitapaa pohdittaessa päädyttiin RAID1-tekniikkaan eli kovalevyjen peilaukseen, joka tallentaa saman datan tässä tapauksessa kahdelle samankokoiselle kovalevyille. Näin toisen kovalevyn rikkoutuessa data säilyy toisella kovalevyllä turvasa. Tietyin väliajoin data myös tallennetaan ulkoiseen muistiin, joten edellinen varmuuskopio on esimerkiksi sähkökatkoksen aikana saatavilla.

Virustorjuntaohjelmaksi palvelimen käyttöjärjestelmässä päädyttiin ilmaiseen avoimen lähdekoodin Clam AntiVirus (ClamAV) -ohjelmaan, joka on suunniteltu havaitsemaan troijalaisia, viruksia ja muita haittaohjelmia. Se on arvioitu parhaimmaksi Linux Ubuntun virustorjuntaohjelmaksi vuonna 2016.

Palomuurin osalta päädyttiin valitsemaan UncomplicatedFirewall (UFW) erityisesti helppokäyttöisyytensä vuoksi. Se on käyttöliittymä, joka on kehitetty käyttämään Ubuntun omaa iptables-palomuuria.

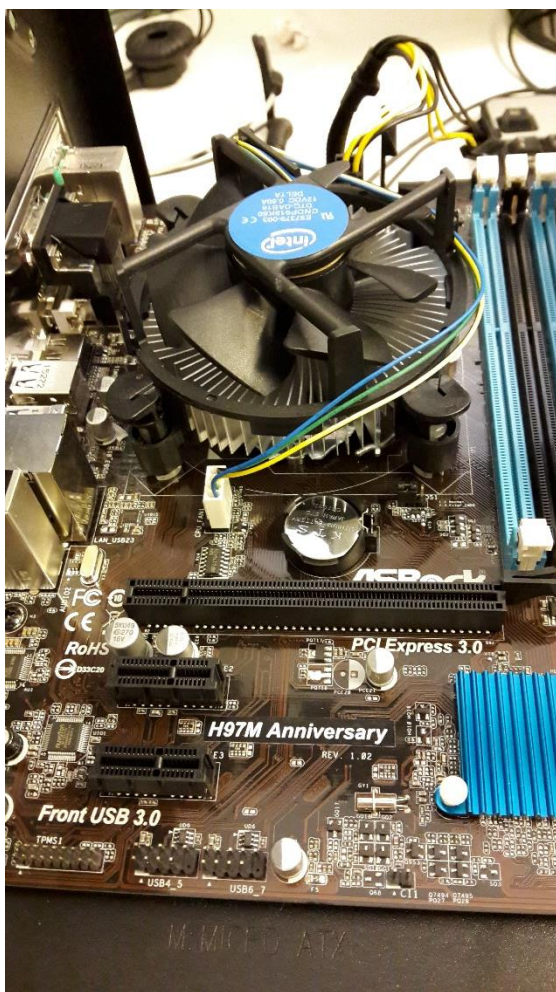
OwnCloud-sovelluksen asennusohjeessa on suositeltu käyttämään tiedonsiirron suojaukseen nykyisin hyvin yleistä HTTPS-protokollaa, joka on HTTP-protokollan ja TLS/SSL-protokollan yhdistelmä. Se oli tässä projektissa selkeä valinta.

### 3 TOTEUTUS

Projektin toteutus lähti liikkeelle palvelinkoneen kokoamisella. Seuraavaksi koneeseen asennettiin Linux Ubuntu 14.04 LTS Server-käyttöjärjestelmä, jonka asennusvaiheessa koneen kahdelle kovalevylle toteutettiin RAID1-tekniikalla kovalevyjen peilaus. Käyttöjärjestelmän asennuksen jälkeen siihen asennettiin graafiseksi käyttöliittymäksi Kubuntu-desktop sekä valmisteluvaiheen PHP- ja Apache2-asennuspaketit. Näiden vaiheiden jälkeen päästiin asentamaan itse ownCloud-ohjelmisto. Lopuksi tehtiin tarvittavat konfiguroinnit Apache2 web-palvelimelle, ownCloud-ohjelmistolle ja asennettiin sekä säädettiin tietoturva kuntoon.

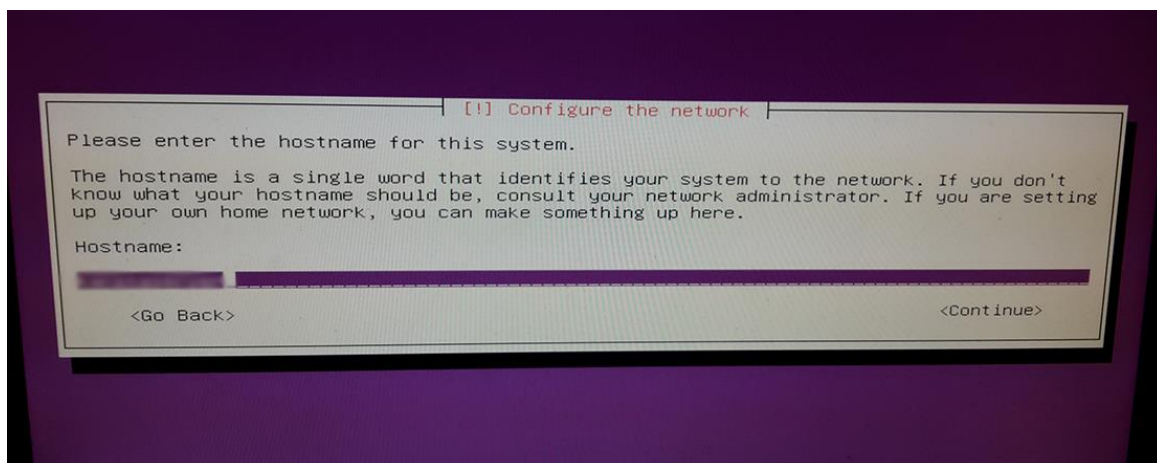
#### 3.1 Palvelinkoneen kokoaminen ja Linux Ubuntu 14.04 LTS Server-käyttöjärjestelmän asennus

Palvelinkoneen kokoamisvaihe oli suhteellisen yksinkertainen, koska kotelo on hyvin tilava ilmanvaihtoa ja jatkokehitystä silmällä pitäen. Muiden komponenttien, kuten prosessorin (kuva 1), liittäminen koteloon asennettuun emolevyyn sujui ongelmitta.



KUVA 1. AsRock H97M Anniversary emolevy asennusvaiheessa

Linux Ubuntu 14.04 LTS Server-käyttöjärjestelmän asennus onnistui USB-muistitikulle tehtyä levykuvaa käyttäen. Asennusvaiheessa ennen RAID1-tekniikan asettamista oli tärkeää antaa palvelimelle yksilöllinen nimi (hostname), koska se todentaa palvelimen muille verkon käyttäjille. (Kuva 2.)



KUVA 2. Yksilöllisen nimen asettaminen palvelimelle käyttöjärjestelmän asennusvaiheessa

### 3.2 Kovalevyjen peilaus RAID1-tekniikalla

Käyttöjärjestelmän asennuksen kohdassa "Partition disks" valitaan ositusmetodiksi "Manual". Seuraavassa kohdassa valitaan ensimmäinen kovalevy (sda), jonka jälkeen vahvistetaan ositus valitsemalla "Yes". Sitten valitaan toinen kovalevy (sdb) ja tehdään sille sama kuin ensimmäiselle levyille. Tämän jälkeen valitaan ensimmäisen levyn tyhjä tila ("pri/log" -alkuinen rivi), jonka jälkeen valitaan tyhjän tilan käyttötavaksi "Create a new partitioning". Osion kooksi asetetaan "4 GB", koska kyseessä on SWAP-osio. Tyypiksi tälle valitaan "Primary" ja sijainniksi "Beginning". Seuraavassa kohdassa käyttötavaksi asetetaan "physical volume for RAID", jonka jälkeen poistutaan osiointiasetuksista valitsemalla "Done setting up the partition".

Ensimmäisen kovalevyn (sda) SWAP-osion luomisen jälkeen valitaan samalle kovalevylle jäänyt tyhjä levytila ja valitaan käyttötavaksi "Create a new partition". Tälle jätetään kooksi loppu levytila, joka on oletuksena. Tyypiksi valitaan "Logical" ja käyttötavaksi "physical volume for RAID". Sitten poistutaan takaisin näkymään, jossa näkyvät molemmat kovalevyt ja niiden osiointit. Aikaisemmin luetellut kohdat toistetaan toiselle kovalevylle, jotta osiointit ovat molemmilla levyillä identtiset.

Kun osiointit ovat molemmilla levyillä samanlaiset, päästään asettamaan peilaus valitsemalla "Configure software RAID". Asennus kysyy vahvistusta, johon vastataan "Yes". Tämän jälkeen valitaan "Create MD device" ja sen tyypiksi "RAID1". Aktiivisten laitteiden lukumääräksi kirjoitetaan levyjen lukumäärä eli "2" ja vapaiden laitteiden lukumääräksi "0". Seuraavaksi valitaan RAID1-taulukkuun molempien levyjen SWAP-osiot (tavallisesti "sda1" ja "sdb1") ja painetaan "Continue". Seuraavaksi toistetaan samalla tavalla kohta "Create MD device" levyjen ROOT-osioille (suurimmat osiot) ja valitaan "Finish".

Nyt asennuksen osiointi-vaiheessa ovat näkyvissä kovalevyjen lisäksi "RAID1 device #0" (SWAP-osio) ja "RAID1 device #1" (ROOT-osio). "RAID1 device #0" alapuolelta valitaan rivi: "#1 4.0 GB" ja sille valitaan käyttötavaksi "swap area". Sitten vastaavasti "RAID device #1":lle valitaan käyttötavaksi "Ext4 journaling file system" ja asetetaan "Mount point":ksi root-kansio ("/"). Sitten viimeistellään asennus valitsemalla levy/osio-näkymässä "Finish partitioning and write changes to disk".

### 3.3 Käyttöjärjestelmän valmistelu tiedostonjakopalvelua varten

Valmisteluvaihe alkoi päivittämällä käyttöjärjestelmä ajantasalle root-käyttäjänä komennoilla "apt-get update" ja "apt-get upgrade". Päivityksen jälkeen käyttöjärjestelmään asennettiin ylläpitokäyttöä helpottamaan graafinen käyttöliittymä Kubuntu Plasma Desktop komennolla "apt-get install kubuntu-desktop", koska palvelinversion käyttöliittymä ei ole oletuksena graafinen. Lisäksi se sisältää Kubuntun suositusohjelmistot, kuten esimerkiksi OpenOfficen.

Seuraavaksi käyttöjärjestelmään asennettiin ownCloud asennusohjeiden suositusten mukaisesti Apache2-, PHP- ja tietokantamoduulit komennoilla:

```
apt-get install apache2 mariadb-server libapache2-mod-php5
apt-get install php5-gd php5-json php5-mysql php5-curl
apt-get install php5-intl php5-mcrypt php5-imagick
```

### 3.4 OwnCloud-tiedostonjakosovelluksen asennus

OwnCloud-sovelluksen asentaminen lähti liikkeelle lataamalla tar.bz2-paketti sen kotisivuilta ja purkamalla se komennolla "tar -xjf owncloud-9.1.0.tar.bz2". Tämän jälkeen kansio kopioitiin web-palvelimen hakemistoon komennolla "cp -r owncloud /var/www".

Seuraavaksi luotiin "owncloud.conf" -asetustiedosto Apache2 web-palvelimelle komennolla "nano /etc/apache2/sites-available/owncloud.conf", johon kirjoitettiin seuraavat asetukset asennusohjeiden mukaisesti:

```
Alias /owncloud "/var/www/owncloud/"
```

```
<Directory /var/www/owncloud/>
```

```
Options +FollowSymlinks
```

```
AllowOverride All
```

```
<IfModule mod_dav.c>
```

```
Dav off
```

```
</IfModule>
```

```
SetEnv HOME /var/www/owncloud
```

```
SetEnv HTTP_HOME /var/www/owncloud
```

```
</Directory>
```

Tämän jälkeen asetustiedosto linkitettiin Apache2:n "sites-enabled" -kansioon komennolla: "ln -s /etc/apache2/sites-available/owncloud.conf /etc/apache2/sites-enabled/owncloud.conf".

Seuraavaksi suoritettiin asennusohjeissa suositellut komennot Apache2 web-palvelimelle, jotta own-Cloud toimisi oikein:

```
a2enmod rewrite
a2enmod headers
a2enmod env
a2enmod dir
a2enmod mime
a2enmod setenvif
```

Sitten lisättiin komento "Satisfy Any" aiempaan "owncloud.conf" -tiedostoon <Directory> -tagien sisälle ja käynnistettiin Apache2 uudestaan komennolla "service apache2 restart", jotta asetukset tulivat voimaan.

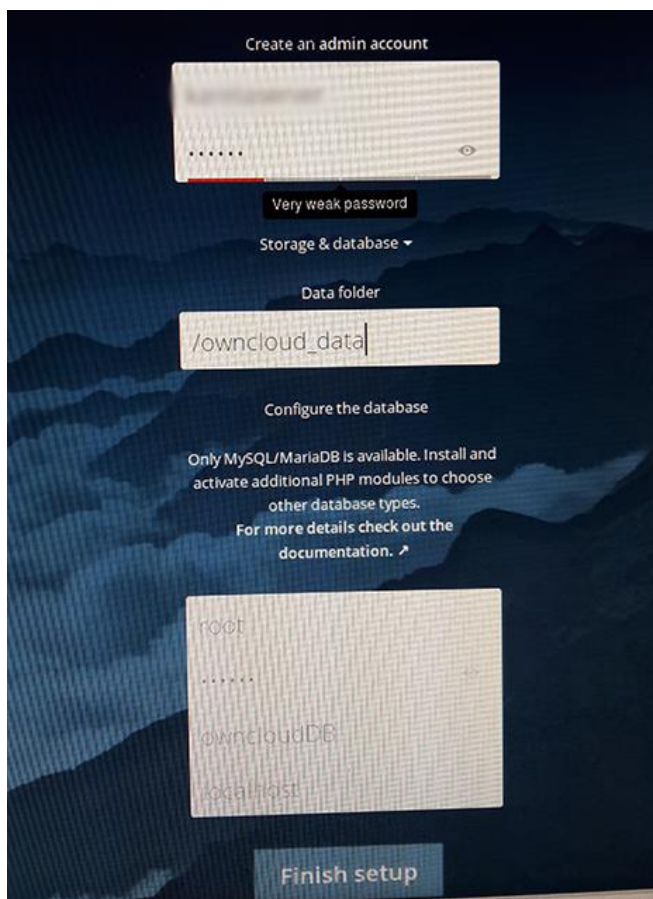
Ennen asennusvelhoa kytkettiin SSL-salaus päälle komennolla:

```
a2enmod ssl
a2ensite default-ssl
service apache2 reload
```

ja annettiin HTTP-käyttäjälle tilapäisesti oikeudet owncloud-kansioon komennolla:

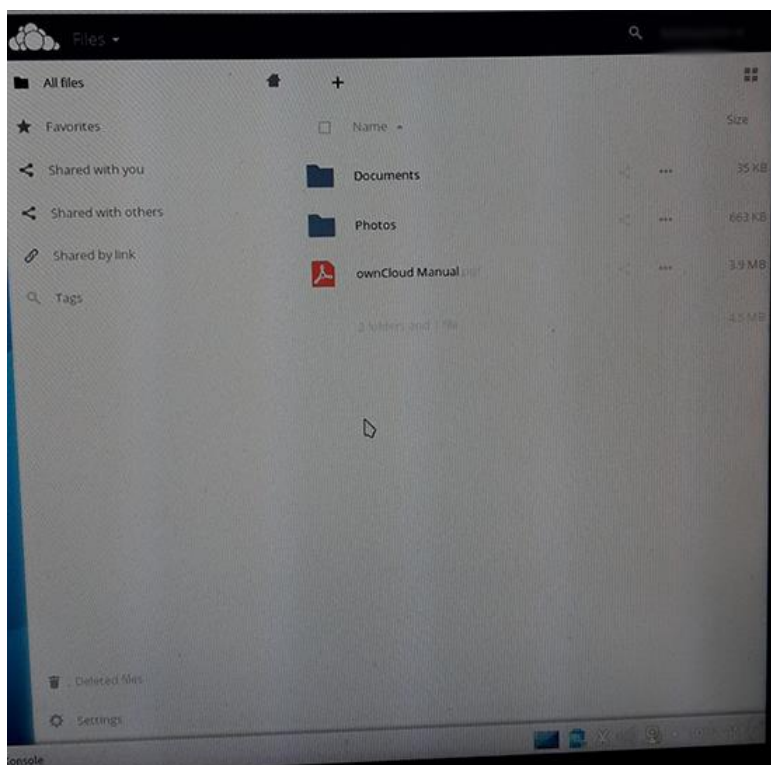
```
mkdir /owncloud_data
chown -R www-data:www-data /owncloud_data/
```

Lopuksi suoritettiin asennus loppuun graafisella asennusvelholla, joka avataan internetselaimella osoitteessa "http://localhost/owncloud" (kuva 3). Siinä määritetään sovellukselle ylläpitäjän käyttäjätunnus, salasana, sovelluksen datakansio ja tietokanta.



KUVA 3. OwnCloud asennusvelho

Lopuksi ownCloud on tietoturvaa ja verkkoyhteyden kautta käyttöä lukuunottamatta valmis. (Kuva 4.)

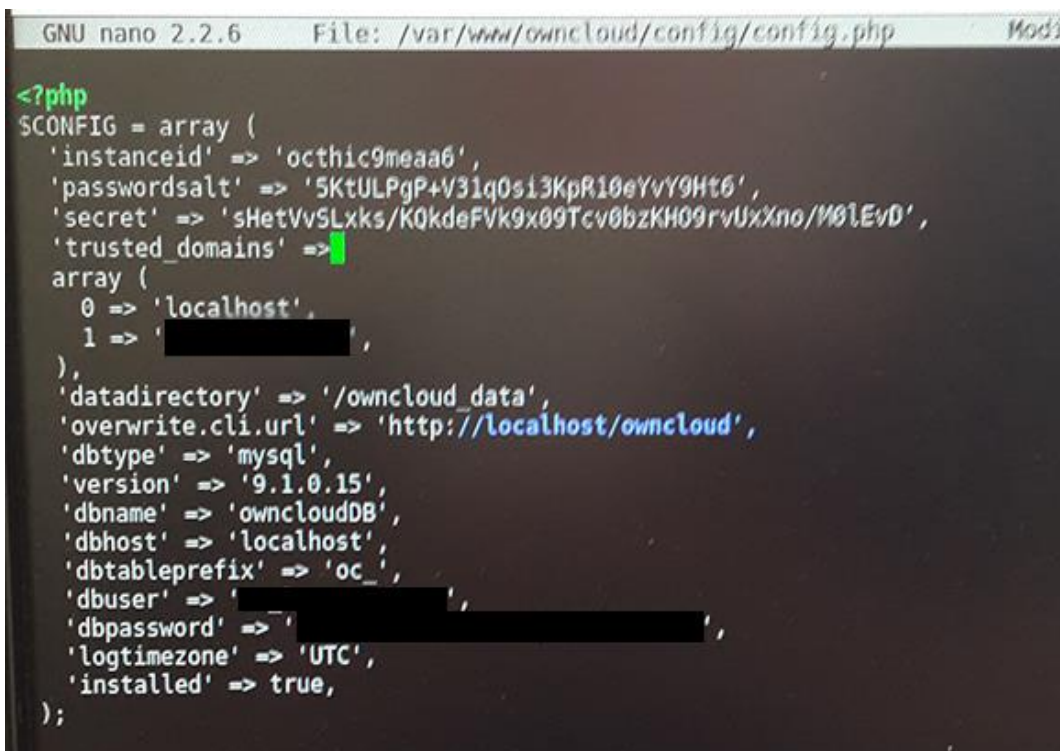


KUVA 4. OwnCloud web-käyttöliittymä



## 3.5 Tietoturva

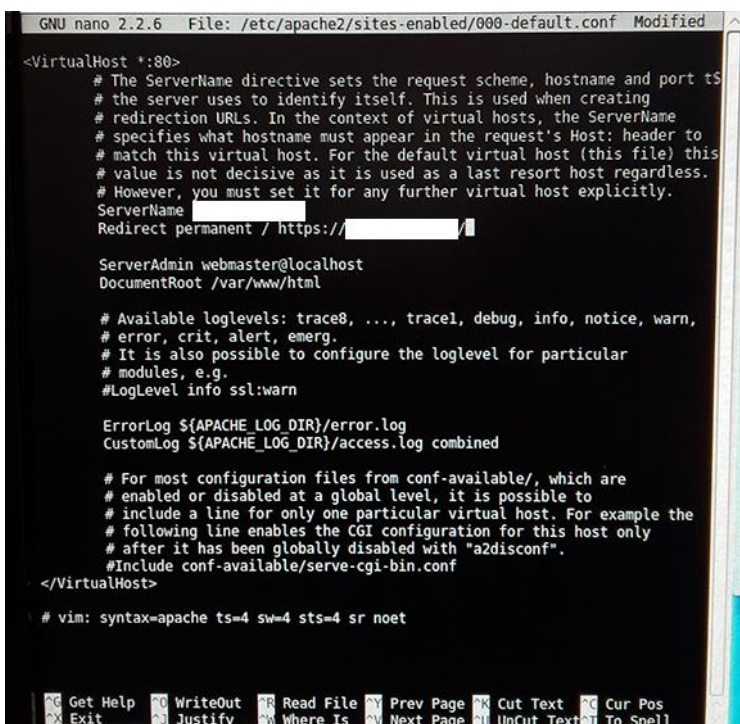
Ensimmäiseksi ownCloud:lle asetettiin palvelimen ulkoinen ip-osoite "Trusted domains" -taulukkoon kohtaan 1 komennolla "nano /var/www/owncloud/config/config.php". (Kuva 5.)



```
GNU nano 2.2.6 File: /var/www/owncloud/config/config.php Modified
<?php
$CONFIG = array (
  'instanceid' => 'octic9meaa6',
  'passwordsalt' => '5KtULPgP+V31q0si3KpR10eYvY9Ht6',
  'secret' => 'sHetVvSLxks/KQkdeFVv9x09Tcv0bzKH09rvUxXno/Y01EvD',
  'trusted_domains' =>
  array (
    0 => 'localhost',
    1 => '...',
  ),
  'datadirectory' => '/owncloud_data',
  'overwrite.cli.url' => 'http://localhost/owncloud',
  'dbtype' => 'mysql',
  'version' => '9.1.0.15',
  'dbname' => 'owncloudDB',
  'dbhost' => 'localhost',
  'dbtableprefix' => 'oc_',
  'dbuser' => '...',
  'dbpassword' => '...',
  'logtimezone' => 'UTC',
  'installed' => true,
);
```

KUVA 5. OwnCloud:n asetustiedosto

Seuraavaksi Apache2 web-palvelimen oletusasetuksiin määriteltiin palvelimelle tuleville HTTP-yhteyksille uudelleenohjaus suojattuun HTTPS-yhteyteen. (Kuva 6.)



```
GNU nano 2.2.6 File: /etc/apache2/sites-enabled/000-default.conf Modified
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port to
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
ServerName ...
Redirect permanent / https://.../

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

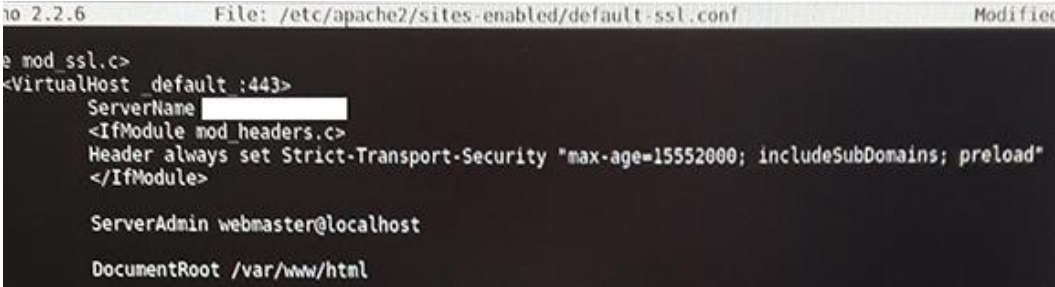
# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

KUVA 6. HTTP-yhteyden uudelleenohjaus suojattuun HTTPS-yhteyteen

Tietoturvan parantamiseksi Apache2:n SSL-oletusasetuksiin lisätään määrittys: (Kuva 7.)

```
<IfModule mod_headers.c>
    Header always set Strict-Transport-Security "max-age=15552000; includeSubDomains"
</IfModule>
```



The screenshot shows a terminal window displaying the configuration for the default SSL virtual host in Apache2. The file path is /etc/apache2/sites-enabled/default-ssl.conf. The configuration includes the following lines:

```

#LoadModule ssl_module modules/mod_ssl.c
LoadModule ssl_module modules/mod_ssl.c
<VirtualHost default :443>
    ServerName [redacted]
    <IfModule mod_headers.c>
        Header always set Strict-Transport-Security "max-age=15552000; includeSubDomains; preload"
    </IfModule>

    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html

```

KUVA 7. Apache2 web-palvelimen SSL-asetukset

Lopuksi rajataan ownCloud kansioden käyttöoikeus ainoastaan HTTP-käyttäjälle ja ryhmälle ajamalla alla oleva skripti:

```
#!/bin/bash
ocpath='/var/www/owncloud'
htuser='www-data'
htgroup='www-data'
rootuser='root'

printf "Creating possible missing Directories\n"
mkdir -p $ocpath/data
mkdir -p $ocpath/assets
mkdir -p $ocpath/updater

printf "chmod Files and Directories\n"
find ${ocpath}/ -type f -print0 | xargs -0 chmod 0640
find ${ocpath}/ -type d -print0 | xargs -0 chmod 0750

printf "chown Directories\n"
chown -R ${rootuser}:${htgroup} ${ocpath}/
chown -R ${htuser}:${htgroup} ${ocpath}/apps/
chown -R ${htuser}:${htgroup} ${ocpath}/assets/
chown -R ${htuser}:${htgroup} ${ocpath}/config/
chown -R ${htuser}:${htgroup} ${ocpath}/data/
chown -R ${htuser}:${htgroup} ${ocpath}/themes/
chown -R ${htuser}:${htgroup} ${ocpath}/updater/
```



```

chmod +x ${ocpath}/occ

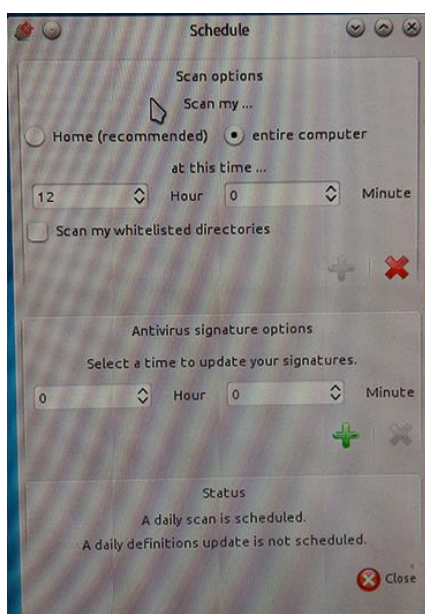
printf "chmod/chown .htaccess\n"
if [ -f ${ocpath}/.htaccess ]
then
  chmod 0644 ${ocpath}/.htaccess
  chown ${rootuser}:${htgroup} ${ocpath}/.htaccess
fi
if [ -f ${ocpath}/data/.htaccess ]
then
  chmod 0644 ${ocpath}/data/.htaccess
  chown ${rootuser}:${htgroup} ${ocpath}/data/.htaccess
fi

```

Nyt palvelimen ownCloud-kansio on käytettävissä vain paikallisesti palvelimelta tai suojatun HTTPS-yhteyden kautta.

Tietoturvan varmistamiseksi palvelimelle asennettiin UncomplicatedFirewall (UFW), joka on Ubuntun oman iptables-palomuurin käyttöliittymä. Asennus tapahtuu terminal-näkymässä komennolla "apt-get install ufw". Sen avulla estetään HTTP-yhteys komennolla "ufw deny http" ja vastaavasti sallitaan HTTPS-yhteys komennolla "ufw allow https".

Virustorjunta puolestaan hoidettiin asentamalla ClamAV-virustorjuntasovellus terminal-näkymässä komennolla "apt-get install clamav clamav-daemon". Seuraavaksi päivitettiin sovelluksen virustietokanta komennolla "freshclam", jonka jälkeen asennettiin sovellukselle graafinen käyttöliittymä ClamTK käyttäen komentoa "apt-get install ClamTK". Lopuksi sovellukselle asetettiin koko tietokoneen päivittäinen ajoitettu tarkistus. (Kuva 8.)



KUVA 8. Ajoitetun virustarkistuksen asettaminen

## 4 TESTAUS

Ensimmäisenä testausvaiheessa yritettiin ottaa palvelimeen HTTP-yhteys eri lähiverkossa olevalta laitteelta menemällä sen internetselaimella palvelimen ulkoiseen ip-osoitteeseen, jolla varmistettiin uudelleenohjauksen toimivuudesta SSL-salattuun HTTPS-yhteyteen. Tässä vaiheessa näkyi Apache2 web-palvelimen oletus etusivu, joten ownCloud web-näkymän avaamiseksi osoiteriville lisättiin ip-osoitteen perään `"/owncloud"`, josta aukesi kirjautumissivu. Tämän toimenpiteen onnistuminen myös vahvisti, että palvelin ylipäätään toimii sille tarkoitetussa tehtävässä.

Seuraavaksi ownCloud-palveluun kirjauduttiin sille määritetyllä ylläpitäjän käyttäjätunnuksella, jolla luotiin kolme käyttäjätunnusta testaustarkoituksiin. Näillä käyttäjätunnuksilla lisättiin palveluun tiedostoja, joita jaettiin muiden käyttäjien kesken erilaisin oikeuksin. Yhdeltä tavalliselta käyttäjältä esimerkiksi piilotettiin tiedosto `"testi.txt"`, jonka luku-oikeus sallittiin toiselle tavalliselle käyttäjälle, mutta ei muokkausoikeutta. Tiedosto käyttäytyi eri käyttäjillä asetusten mukaisesti.

OwnCloud-sovelluksen toimivuuden varmistuttua yritettiin avata ownCloud:n data-kansio paikallisesti palvelinkoneen tiedostonhallintaohjelmalla. Kansio ei auennut, koska oikeus kansioon oli asetettu vain `www-data` -käyttäjälle. Sen vuoksi kansiota ei pääse tarkastelemaan paikallisesti muokkaamatta sen käyttöoikeuksia `root`-käyttäjänä, mikä parantaa tietoturvaa, jos palvelinkone esimerkiksi varastetaan. Lisäksi käyttöjärjestelmä on luonnollisesti lukittu salasanalla. Ollessaan käyttämättömänä puoli tuntia järjestelmä kysyy salasanaa.

Loppuvaiheessa jokaiselle tiedostonjakopalvelun tavalliselle loppukäyttäjälle tehtiin käyttäjätunnukset ja ohjeistettiin heitä vaihtamaan salasanansa. Palvelun ylläpitokäyttäjälle ohjeistettiin ylläpidolliset perusasiat kuten esimerkiksi, miten tavallisen käyttäjän salasana nollataan tarvittaessa. Lopuksi käyttäjien Windows-päätteille asennettiin palvelun synkronointisovellus, jotta tiedostot ovat käytettävissä Windows-resurssienhallinnan kautta.

## 5 JATKOKEHITYS

Jatkokehitysideoista ensimmäisenä esille nousivat SSL-sertifikaatin hankkiminen sekä välimuistin konfigurointi tiedostonjakopalveluun. SSL-sertifikaatti eli SSL-varmenne on vahvistus palvelun käyttäjälle, että palvelu on turvallinen. Ilman SSL-sertifikaattia internetselain antaa ilmoituksen sivustolle siirryttäessä, että sivusto ei ole turvallinen. Varmenne ei kuitenkaan projektin tarkoitukseen ollut välttämätön, koska palvelu ei ole ulkopuolisille suunnattu. Välimuistin (memory cache) konfigurointi ownCloud palveluun on tiedostojen siirtonopeuden hienoista parantamista, joten se ei ole palvelun käytölle välttämätöntä ainakaan toistaiseksi.

Kaukaisempia kehitysideoita itse palvelinta ajatellen ovat tehtävien, kuten web- ja sähköpostipalveluiden lisääminen sille.

## 6 YHTEENVETO

Projekti oli kokonaisuudessaan mielenkiintoinen ja suhteellisen haastava, mutta mikä tärkeintä, itse pilvitallennuspalvelun käyttäjät olivat lopputulokseen tyytyväisiä. Haastavuutta projektin lähtötilanteessa lisäsivät seikat, että yrityksellä ei ole toistaiseksi palvelinta, verkkoyhteys on toteutettu ADSL modeemin kautta sekä yrityksessä ei ollut kovin suurta osaamista tietoverkkotekniikasta. Projekti oli siis todella kokonaisvaltainen, koska siihen sisältyi palvelinkoneen suunnittelu komponenteista lähtien, pilvitallennusohjelmiston valinta sekä tietoturvan suunnittelu. Palvelimen kytkeminen verkkoon onnistui modeemin yhden LAN-portin siltaamisella, joten porttiohjausmentelmän käytöltä vältyttiin.

## LÄHTEET

The ownCloud developers. ownCloud Features. Viitattu 29.5.2017. <https://owncloud.org/features/>

The ownCloud developers. ownCloud 9.1 Server Administration Manual. Viitattu 29.5.2017. [https://doc.owncloud.org/server/latest/admin\\_manual/](https://doc.owncloud.org/server/latest/admin_manual/)

Leurs, Laurens. 2017. RAID. Viitattu 29.5.2017. <https://www.prepressure.com/library/technology/raid>

Cisco and/or its affiliates. ClamAV. Viitattu 29.5.2017. <https://www.clamav.net/>

Symantec Corporation. What is SSL, TLS and HTTPS?. Viitattu 29.5.2017. <https://www.symantec.com/page.jsp?id=ssl-information-center>

Computer Hope. HTTP. Viitattu 29.5.2017. <https://www.computerhope.com/jargon/h/http.htm>

Linux.fi-wiki. Swap. Viitattu 30.5.2017. <https://www.linux.fi/wiki/Swap>