

Debian Jessie Owncloud palvelin

Linux Servers

Joni Korpihalkola

Harjoitustyö

11/16

TTTS0400

Tieto- ja viestintätekniikka

Sisällysluettelo

Kuvioluettelo	2
1 Johdanto	4
2 Owncloud asennus	4
3 Owncloud asetukset	6
4 Owncloud suojauksen vahventaminen	9
4.1 Owncloudin kansioden oikeudet	9
4.2 HTTPS uudelleenohjaus ja itseallekirjoitettu sertifikaatti	10
4.3 Palomuuuri	12
4.4 Virustorjunta	14
4.5 Config.php muokkaaminen	15
5 Owncloud optimointi	16
5.1 Välimuistin asettaminen	16
5.2 Tausta-ajo	17
6 Toimivuuden testaus	17
6.1 HTTPS uudelleenohjaus	17
6.2 Tiedostojen vienti palvelimelle ja jakaminen	18
7 Kehitettävää tulevaisuudessa	20
7.1 Käyttö sisäverkon ulkopuolelta	20
7.2 SELinux	21

Kuvioluettelo

Kuvio 1. Komento, jolla repositorio lisätään lähteisiin.	4
Kuviot 2-3. Latasin owncloud repositorion release avaimen ja lisäsin sen pakettilistaan..	4
Kuvio 4. Owncloudin asennus.	5
Kuvio 5. Owncloudin käyttöliittymä.	5
Kuvio 6. Virtuaalikoneen virtuaalilevyt.	6
Kuviot 7-11. Owncloud-data kansion valmistelemineen.	7
Kuvio 12. Owncloud-data kansion omistajan vaihtaminen.	7
Kuvio 13. Owncloudin asennuksen viimeistely.	8
Kuvio 14. Ajettava skripti.	9
Kuvio 15. Skripti ajettu.	10
Kuvio 16. Komento, jolla apachen ssl-moduulin saa päälle.	10
Kuviot 17-20. Komennot sertifikaatin ja avaimen luomiseen, sekä niiden siirtäminen ssl kansioon.	11
Kuvio 21. Lisätyt rivit .htaccess tiedostoon.	11
Kuvio 22. VirtualHost kuuntelee liikennettä kaikista osoitteista 443 porttiin ja sille on kerrottu sertifikaatin ja avaimen polut.	11
Kuvio 23. Komento, jolla headers-moduuli laitetaan päälle.	11
Kuvio 24. Uudet konfiguraatiot VirtualHostin sisällä korostettu keltaisella.	12
Kuvio 25. Iptables säännöt tiedostossa.	12

Kuviot 26-27. Komennot, jolla iptables säännöt haetaan tiedostosta, voimassa olevat iptables säännöt tulostetaan ja säännöt tallennetaan iptables master- tiedostoon.	13
Kuviot 28-29. Komento, joka hakee käynnistyksen yhteydessä iptables säännöt iptables2.rules tiedostosta ja komento, jolla annetaan skriptille ajo-oikeudet.	13
Kuviot 30-32. Komennot Clamav asennukseen, käynnistykseen ja käynnistykseen palvelimen käynnistyksen yhteydessä.	14
Kuvio 33. Antivirus App Owncloudin sivulla.....	14
Kuviot 34-35. Lokitason määrittäminen ja antiviruksen konfigurointi admin-sivulla.....	15
Kuvio 36. ClamAV socketin sijainti.	15
Kuvio 37. Komento, joka näyttää php version.	16
Kuvio 38. Admin sivun Cron asetukset.....	17
Kuvio 39. HTTPS uudelleenohjaus ja sertifikaatti.....	18
Kuvio 40. Testikäyttäjän luonti.....	18
Kuvio 41. Näkymä kännykälläni, kun vein palvelimelle uuden tiedoston.....	19
Kuvio 42. Kuvan lataaminen jaetusta linkistä	20
Kuviot 43-44. Make komennot ja make load virheilmoitus.....	21

1 Johdanto

Tehtävänä oli luoda Linux-pohjainen palvelin ottaen huomioon, mitä tietoturva asetuksia palvelimelle kannattaisi laittaa. Päätin tehdä Owncloud palvelun, jonka avulla pystyy tallentamaan palvelimelle tiedostoja. Tiedostoja voi jakaa muille linkkien avulla tai ladataan toiselle koneelle. Owncloud palvelinta pyörittää Windows 10 pöytäkoneellani virtuaalikoneeseen asennettu Debian 8.6 eli "Jessie" käyttöjärjestelmä. Asennetun Owncloudin versio on 8.1.9.

2 Owncloud asennus

Ensiksi lisäsin Owncloudin virallisen repositorion käyttöjärjestelmäni repositorio lähteisiin pääkäyttäjänä:

```
root@Debian:/home/joni# echo 'deb http://download.opensuse.org/repositories/isv:/ownCloud:/community/Debian_8.0/ /' >> /etc/apt/sources.list.d/owncloud.list
```

Kuvio 1. Komento, jolla repositorio lisätään lähteisiin.

```
root@Debian:/tmp# wget http://download.opensuse.org/repositories/isv:/ownCloud:/community/Debian_8.0/Release.key
```

```
root@Debian:/tmp# apt-key add - < Release.key  
OK
```

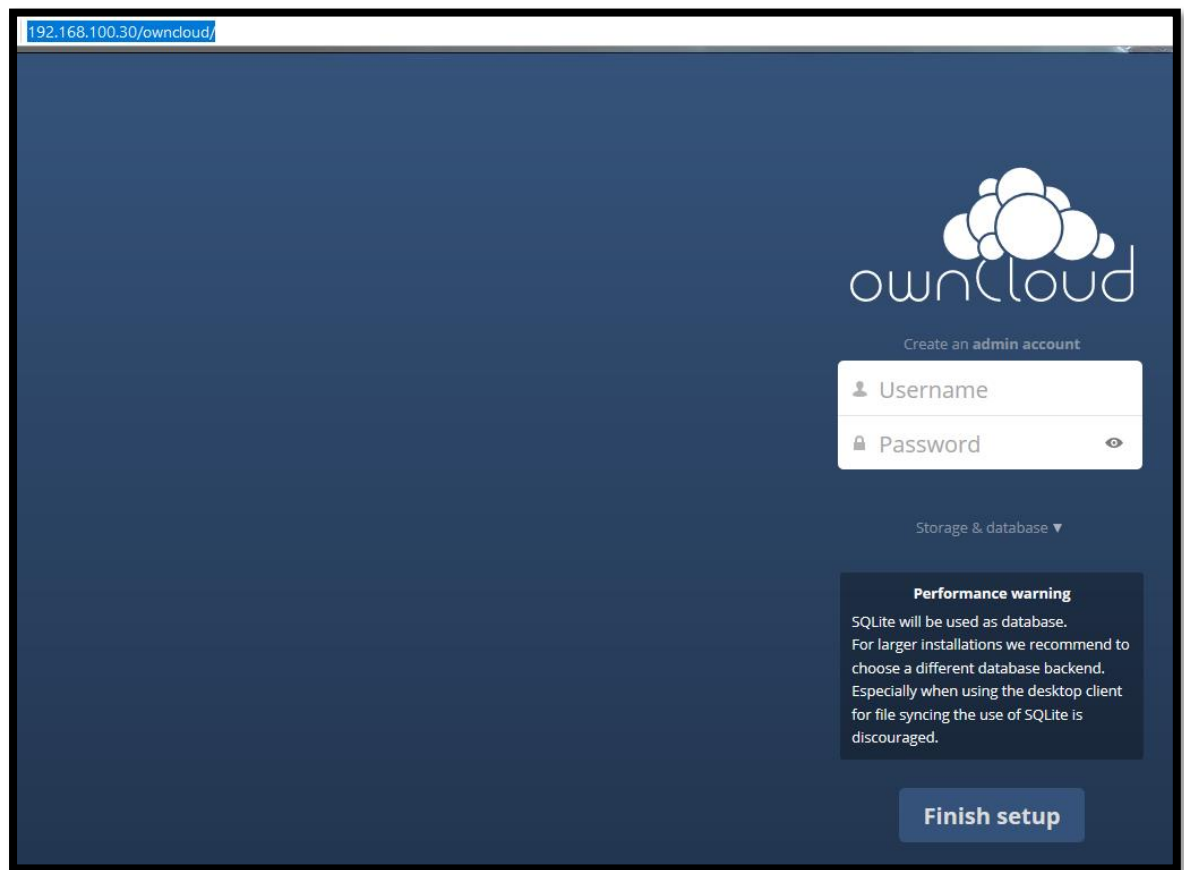
Kuviot 2-3. Latasin owncloud repositorion release avaimen ja lisäsin sen pakettilistaan.

Nyt kun owncloudin repositorio on lisätty, sen pystyy asentamaan ja se asentaa samalla kaikki tarvittavat paketit, mitä se käyttää, kuten php5 ja tarvittavat fontit.

```
root@Debian:/tmp# apt-get install owncloud
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  fonts-font-awesome fonts-linuxlibertine fonts-lohit-deva fonts-wqy-microhei
  javascript-common libapache2-mod-php5 libcurl3 libjs-chosen libjs-dojo-core
  libjs-dojo-dijit libjs-dojo-dojox libjs-jcrop libjs-jquery
  libjs-jquery-metadata libjs-jquery-minicolors libjs-jquery-mousewheel
  libjs-jquery-tablesorter libjs-jquery-timepicker libjs-jquery-ui
  libjs-mediaelement libjs-pdf libjs-sphinxdoc libjs-twitter-bootstrap
```

Kuvio 4. Owncloudin asennus.

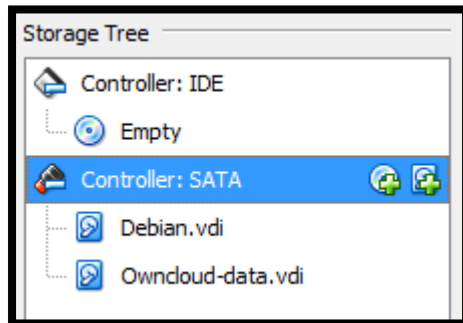
Asennuksen jälkeen pöytäkoneen selaimella pääsen Owncloudin HTML käyttöliittymään kirjoittamalla osoiteriville 192.168.110.30/owncloud, jossa IP-osoite on Debian virtuaalikoneeni osoite.



Kuvio 5. Owncloudin käyttöliittymä.

3 Owncloud asetukset

Owncloudin dokumentaatio suosittelee, että Owncloudin käyttämä datakansio kannattaa sijoittaa /var/www:n ulkopuolelle tietoturvasyistä. Tein ensin datakansiolle oman virtuaalilevyn nimeltä Owncloud-data ja loin /var/owncloud-data kansion.



Kuvio 6. Virtuaalikoneen virtuaalilevyt.

Tein uudelle levylle maksimikokoisen osituksen, alustin sen ext4-tiedostojärjestelmällä ja liitin sen uuteen owncloud-data kansioon muokkaamalla fstab tiedostoa.

```
root@Debian:/home/joni# fdisk /dev/sdb
```

```
root@Debian:/sbin# mkfs.ext4 /dev/sdb1
mke2fs 1.42.12 (29-Aug-2014)
Creating filesystem with 2621184 4k blocks and 655360 inodes
Filesystem UUID: 10059014-9708-4647-a305-649bd975f467
```

```
root@Debian:/var# mkdir /var/owncloud-data
```

```

GNU nano 2.2.6          File: /etc/fstab          Modified
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>          <dump> <pass>
# / was on /dev/sda1 during installation
UUID=bc41e01f-fad5-4ecc-a6bd-dfbf925b4d7c /          ext4      errors=remoun$
# swap was on /dev/sda5 during installation
UUID=ebd7fcc0-1998-477e-b776-b5560dfaace3 none      swap      sw          $
/dev/sr0          /media/cdrom0    udf,iso9660 user,noauto    0          0
/dev/sr1          /media/cdrom1    udf,iso9660 user,noauto    0          0
/dev/sr2          /media/cdrom2    udf,iso9660 user,noauto    0          0
/dev/sr3          /media/cdrom3    udf,iso9660 user,noauto    0          0
/dev/sdb1         /var/owncloud-data ext4      defaults      0          0

```

```
root@Debian:/var# mount owncloud-data
```

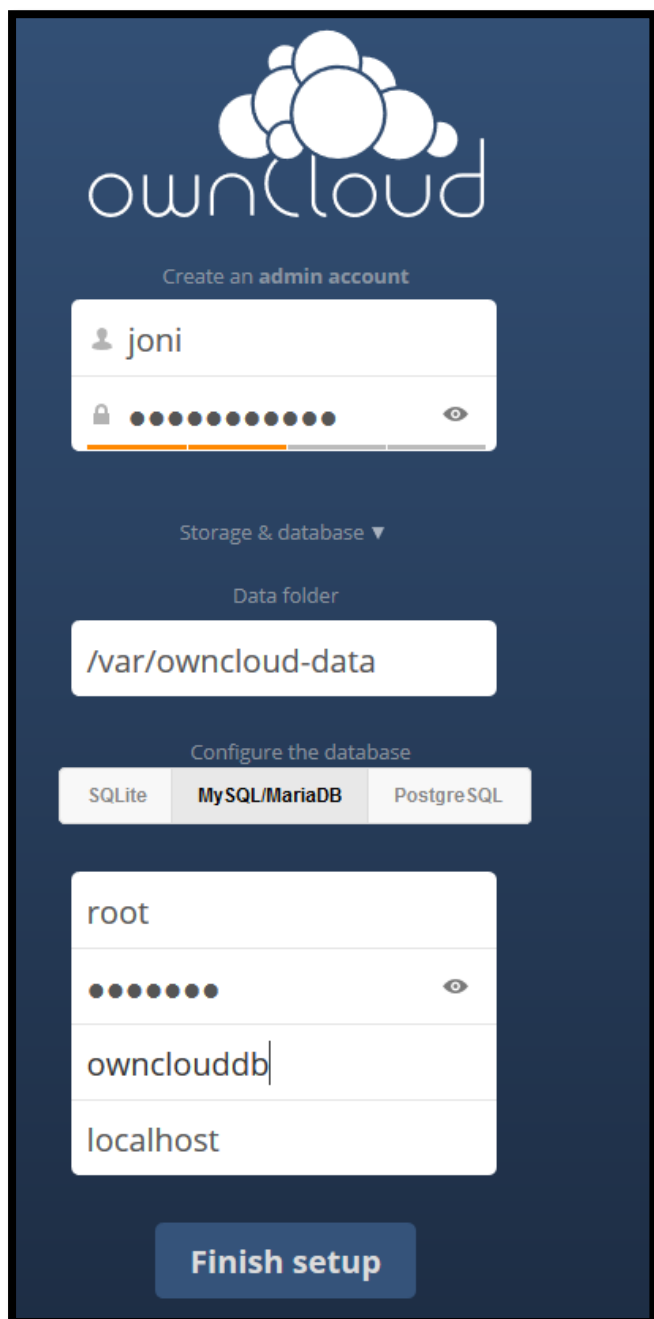
Kuviot 7-11. Owncloud-data kansion valmistelemine.

Owncloud dokumentaation mukaan data kansion täytyy olla HTTP käyttäjän ja ryhmän omistuksessa. Debianissa HTTP käyttäjän ja ryhmän nimi on www-data.

```
root@Debian:/var# chown -R www-data:www-data /var/owncloud-data
```

Kuvio 12. Owncloud-data kansion omistajan vaihtaminen.

Nyt kun datakansion on luotu ja oikeudet ovat kunnossa, ajoin Owncloudin asennuksen loppuun selaimen käyttöliittymällä. Owncloud pyytää tekemään admin-tilin ja määrittämään datakansion sijainnin. Owncloud osaa luoda itse käyttäjille MySQL tietokannat, joittenka nimi on johdettu Owncloudin käyttäjistä.



The image shows the OwnCloud installation interface on a dark blue background. At the top is the OwnCloud logo, consisting of a cluster of white circles above the text 'ownCloud'. Below the logo is the heading 'Create an admin account'. There are two input fields: the first contains the username 'joni', and the second is a password field with a lock icon on the left and an eye icon on the right. Below these fields is a section titled 'Storage & database' with a downward arrow. Under this, there is a 'Data folder' section with a text box containing '/var/owncloud-data'. Below that is a 'Configure the database' section with three tabs: 'SQLite', 'MySQL/MariaDB' (which is selected and highlighted), and 'PostgreSQL'. Below the tabs are three more input fields: the first contains 'root', the second is a password field with a lock icon on the left and an eye icon on the right, the third contains 'ownclouddb', and the fourth contains 'localhost'. At the bottom of the form is a large blue button with the text 'Finish setup'.

ownCloud

Create an admin account

joni

Storage & database ▼

Data folder

/var/owncloud-data

Configure the database

SQLite MySQL/MariaDB PostgreSQL

root

ownclouddb

localhost

Finish setup

Kuvio 13. Owncloudin asennuksen viimeistely.

4 Owncloud suojauksen vahventaminen

4.1 Owncloudin kansioden oikeudet

Owncloudin dokumentaatiosta löytyy ohjeet vahvojen oikeuksien määrittämiseen, jossa ajetaan alla oleva skripti. Skripti vaihtaa Owncloudin kansioden oikeudet ja luo puuttuvia kansioita.

```
GNU nano 2.2.6      File: owncloud-permissions.sh      Modified
#!/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

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}/config/
chown -R ${htuser}:${htgroup} ${ocpath}/data/
chown -R ${htuser}:${htgroup} ${ocpath}/themes/
chown -R ${htuser}:${htgroup} ${ocpath}/assets/

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
```

Kuvio 14. Ajettava skripti.

```

root@Debian:/# /root/owncloud-permissions.sh
Creating possible missing Directories
chmod Files and Directories
chown Directories
chmod/chown .htaccess

```

Kuvio 15. Skripti ajettu.

4.2 HTTPS uudelleenohjaus ja itseallekirjoitettu sertifikaatti

HTTP ei salaa käyttäjien lähettämiä tietoja. Jos joku esimerkiksi kirjautumisen yhteydessä kaappaa käyttäjän lähettämiä paketteja, hän voi helposti napata salasanan paketeista. Owncloudin dokumentaatio suosittelee, että http liikenne uudelleenohjataan HTTPS:ksi, joka salaa tiedot. Ensiksi täytyy laittaa päälle apachesta ssl moduuli päälle.

```

root@Debian:/home/joni# a2enmod ssl

```

Kuvio 16. Komento, jolla apachen ssl-moduulin saa päälle.

Tämän jälkeen loin itseallekirjoitetun sertifikaatin ja julkisen avaimen,

```

root@Debian:~# openssl genrsa -out owncloud.key 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)

```

```

root@Debian:~# openssl req -new -key owncloud.key -out owncloud.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:

```

```
root@Debian:~# openssl x509 -req -days 365 -in owncloud.csr -signkey owncloud.key
y out owncloud.crt
Signature ok
subject=/C=FI/ST=Keski-Suomi/L=Jyvaskyla/O=JoninPilviPalvelut/OU=PilviOsasto/CN=
joninpilvi/emailAddress=joni.korpihalkola@gmail.com
Getting Private key
```

```
root@Debian:~# mv owncloud.crt /etc/ssl/certs
root@Debian:~# mv owncloud.key /etc/ssl/certs/
```

Kuviot 17-20. Komennot sertifikaatin ja avaimen luomiseen, sekä niiden siirtäminen ssl kansioon.

Uudelleenohjaus HTTP:stä HTTPS:ään alkoi toimimaan, kun muokkasin polussa /var/www/owncloud .htaccess tiedostoon alla mainitut rivit.

```
RewriteEngine On
RewriteCond %{HTTPS} !=on
RewriteRule ^/?(.*) https://%{SERVER_NAME}/owncloud [R,L]
```

Kuvio 21. Lisätyt rivit .htaccess tiedostoon.

Jotta apache käyttäisi aikaisemmin luotuja sertifikaatteja ja avainta, kävin muokkaamassa polussa /etc/apache2/sites-available oletuskonfiguraatiota.

```
<VirtualHost *:443>
    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/owncloud.crt
    SSLCertificateKeyFile /etc/ssl/certs/owncloud.key
</VirtualHost>
```

Kuvio 22. VirtualHost kuuntelee liikennettä kaikista osoitteista 443 porttiin ja sille on kerrottu sertifikaatin ja avaimen polut.

Laitoin päälle vielä HTTP Strict Transport Security headerit, joka estää sertifikaatin hylkäämisen ja pääsyn alisivuille pelkällä HTTP:llä. Ensiksi täytyy laittaa apachen headers moduuli päälle.

```
root@Debian:/var/www/owncloud/config# a2enmod headers
Enabling module headers.
```

Kuvio 23. Komento, jolla headers-moduuli laitetaan päälle

Palasin takaisin aiemmin muokattuun apachen oletuskonfiguraatioon, jossa laitoin VirtualHost määrittelyn sisälle alla olevat rivit.

```
<VirtualHost *:443>
    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/owncloud.crt
    SSLCertificateKeyFile /etc/ssl/certs/owncloud.key
    <IfModule mod_headers.c>
        Header always set Strict-Transport-Security "max-age=15768000; includeSubDomains; preload"
    </IfModule>
</VirtualHost>
```

Kuvio 24. Uudet konfiguraatiot VirtualHostin sisällä korostettu keltaisella.

4.3 Palomuuuri

Debian käyttää oletuspalomuurina iptables palomuuria. Oletuksena se minulla salli kaiken liikenteen. Etsin netistä ohjeet minimaaliseen palomuurikonfiguraatioon. Tein tiedoston iptables2.rules, johon kirjoitin alla olevat säännöt

```
GNU nano 2.2.6      File: /etc/iptables2.rules

*filter

# Accepts all established inbound connections
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Allow all loopback traffic
-A INPUT -i lo -j ACCEPT

# Allow HTTP and HTTPS
-A INPUT -p tcp --dport 80 -j ACCEPT
-A INPUT -p tcp --dport 443 -j ACCEPT
-A INPUT -p udp --dport 80 -j ACCEPT
-A INPUT -p udp --dport 443 -j ACCEPT

# Reject all other inbound
-A INPUT -j REJECT --reject-with icmp-host-unreachable
-A FORWARD -j REJECT

COMMIT
```

Kuvio 25. Iptables säännöt tiedostossa.

Sääntöjen perusteella palomuuuri sallii kaiken loopback liikenteen, perustetut yhteydet ja HTTP sekä HTTPS portit on avattu. Muu liikenne kielletään. Tämän jälkeen laitoin iptables säännöt voimaan.

```
root@Debian:/home/joni# iptables-restore < /etc/iptables2.rules
root@Debian:/home/joni# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination           state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere              anywhere
ACCEPT     tcp  --  anywhere              anywhere              tcp dpt:http
ACCEPT     tcp  --  anywhere              anywhere              tcp dpt:https
ACCEPT     udp  --  anywhere              anywhere              udp dpt:http
ACCEPT     udp  --  anywhere              anywhere              udp dpt:https
REJECT     all  --  anywhere              anywhere              reject-with icmp-host-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination           reject-with icmp-port-unreachable

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

```
root@Debian:/home/joni# iptables-save > /etc/iptables2.rules
```

Kuviot 26-27. Komennot, jolla iptables säännöt haetaan tiedostosta, voimassa olevat iptables säännöt tulostetaan ja säännöt tallennetaan iptables master- tiedostoon.

Tämän jälkeen testasin, että sivu vielä toimii. Tämän jälkeen täytyy asettaa iptables säännöt tulemaan voimaan myös uudelleenkäynnistyksen jälkeen. Loin uuden ajettavan skriptin, määritin sisälle alla olevan komennon ja annoin sille ajo-oikeudet.

```
GNU nano 2.2.6      File: /etc/network/if-pre-up.d/iptables
#!/bin/sh
/sbin/iptables-restore < /etc/iptables2.rules
```

```
root@Debian:/home/joni# chmod +x /etc/network/if-pre-up.d/iptables
```

Kuviot 28-29. Komento, joka hakee käynnistyksen yhteydessä iptables säännöt iptables2.rules tiedostosta ja komento, jolla annetaan skriptille ajo-oikeudet.

4.4 Virustorjunta

Kerta Owncloudin käyttäjät voivat laittaa palvelimelle vapaasti tiedostoja, on hyvä laittaa palvelimelle pyörimään virus skannaukset tartuntojen varalta. Asensin ClamAV:n sekä laitoin sen päälle ja myös käynnistymään järjestelmän käynnistyksen yhteydessä.

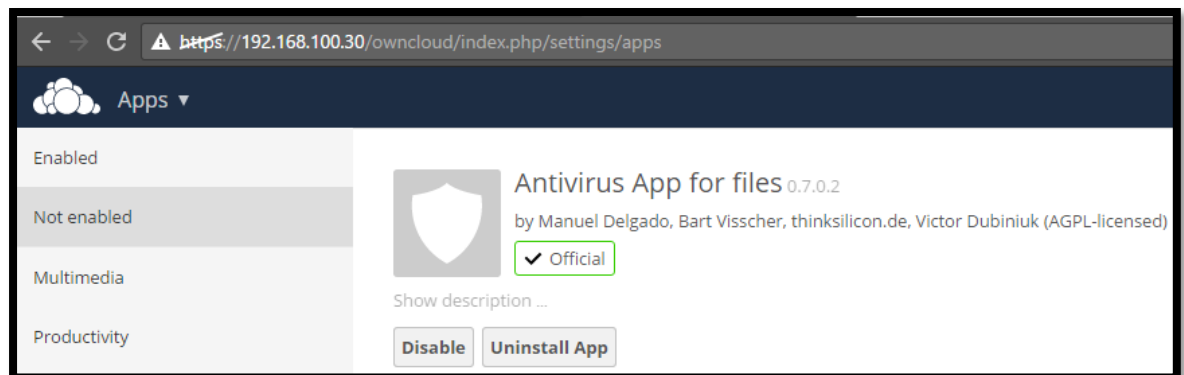
```
root@Debian:/home/joni# apt-get install clamav clamav-daemon
```

```
root@Debian:/etc/clamav# service clamav-daemon start
```

```
root@Debian:/etc/clamav# systemctl enable clamav-daemon
```

Kuviot 30-32. Komennot Clamav asennukseen, käynnistykseen ja käynnistykseen palvelimen käynnistyksen yhteydessä.

Owncloudin Apps - sivuilta löytyi Antivirus App for files- virallinen lisäosa, jonka laitoin päälle.



Kuvio 33. Antivirus App Owncloudin sivulla.

Seuraavaksi menin Owncloudin Admin sivulle ja laitoin Owncloudin pitämään lokia kaikista varoituksista, virheistä ja muista tarpeellisista informaatioista. Admin sivulla pystyy myös konfiguroimaan virustorjuntaa.

Admin ▾

Sharing

Server-side encryption

Log

Log level **Everything (fatal issues, errors, warnings, info, del** ▾

Antivirus Configuration

Mode **Daemon (Socket** ▾

Socket **/var/run/clamav/clam** **Clamav Socket.**

Stream Length **1024** bytes

Action for infected files found while scanning **Delete file** ▾

Save

Kuviot 34-35. Lokitason määrittäminen ja antiviruksen konfigurointi admin-sivulla

Owncloudin dokumentaatiossa suositellaan antiviruksen tilaksi Daemon (Socket), jolloin ClamAV daemon pyörii taustalla vähäisellä prosessorin kulutuksella, kun tiedostoja ei laiteta palvelimelle. Viruksentorjunta poistaa heti saastuneet tiedostot. Socketin sijainnin pystyy varmistamaan alla olevalla komennolla.

```
root@Debian:/etc/clamav# netstat -a | grep clam
unix 2      [ ACC ]     STREAM  LISTENING  23199      /var/run/clamav/clamd
        .ctl
```

Kuvio 36. ClamAV socketin sijainti.

4.5 Config.php muokkaaminen

Apache serverien turvallisuuden kannalta on tärkeää, että .htaccess tiedostoon voi kirjoittaa ja se toimii. Tämän voi tarkistaa kirjoittamalla /var/www/owncloud/config polussa sijaitsevaan config.php:seen rivi: 'check_for_working_htaccess' => true.

Kuvien esikatselu Owncloudissa käyttää PHP:n C:llä kirjoitettuja kirjastoja, jotka ovat potentiaalisesti alttiita hyökkäykselle. Otin toiminnon pois päältä kirjoittamalla rivin: `'enable_previews' => false`.

Lokitiedosto oletuksena käyttää UTC-aikavyöhykettä, kun se ottaa tapahtumien ajat ylös. Vaihdoin tämän suomen aikaan kirjoittamalla: `'logtimezone' => 'Europe/Helsinki'`

Laitoin vielä SSL:n päälle config.php:stä myös riveillä: `'forcessl' => true` ja `'forceSSLforSubdomains' => true`. Nämä uudelleenohjaavat kaiken HTTP liikenteen HTTPS:ksi myös Owncloudin alisivuilla.

5 Owncloud optimointi

5.1 Välimuistin asettaminen

Oletuksena Owncloudissa ei ole määritelty välimuistia. Välimuistin määrittäminen nopeuttaa palvelimen toimintaa, esimerkiksi usein ladatut tiedostot on nopeampi noutaa, jos ne ovat väliaikaisesti tallennettu välimuistiin. Owncloudin dokumentaation mukaan välimuistin määrittämiseen vaihtoehtoja on APC, APCu, Memcached ja Redis. Näistä paikallisesti toimivia ovat APC ja APCu. APCu vaatii PHP 5.5 tai uudemman version. Varmistin, että PHP moduulini versio on tarpeeksi uusi.

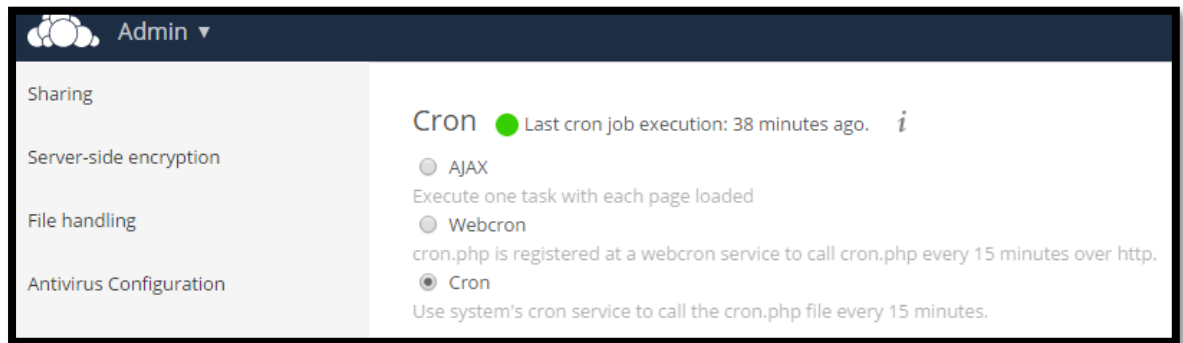
```
root@Debian:/# php -v
PHP 5.6.27-0+deb8u1 (cli) (built: Oct 15 2016 15:53:28)
```

Kuvio 37. Komento, joka näyttää php version.

APCu:n tarvitsema php5-apcu paketti oli asentunut jo Owncloudin asennuksen yhteydessä, joten tarvitsi vain muokata config.php:en yksi rivi: `'memcache.local' => '\OC\Memcache\APCu'`.

5.2 Tausta-ajo

Oletuksena Owncloud käyttää AJAX – metodia taustaohjelmien ajon aikataulutukseen. Tämä metodi on kuitenkin kaikista epäluotettavin, ja Owncloudin dokumentaatio suosittelee metodin vaihtamista Croniin. Asetusta voi muuttaa Owncloudin Admin sivuilla.

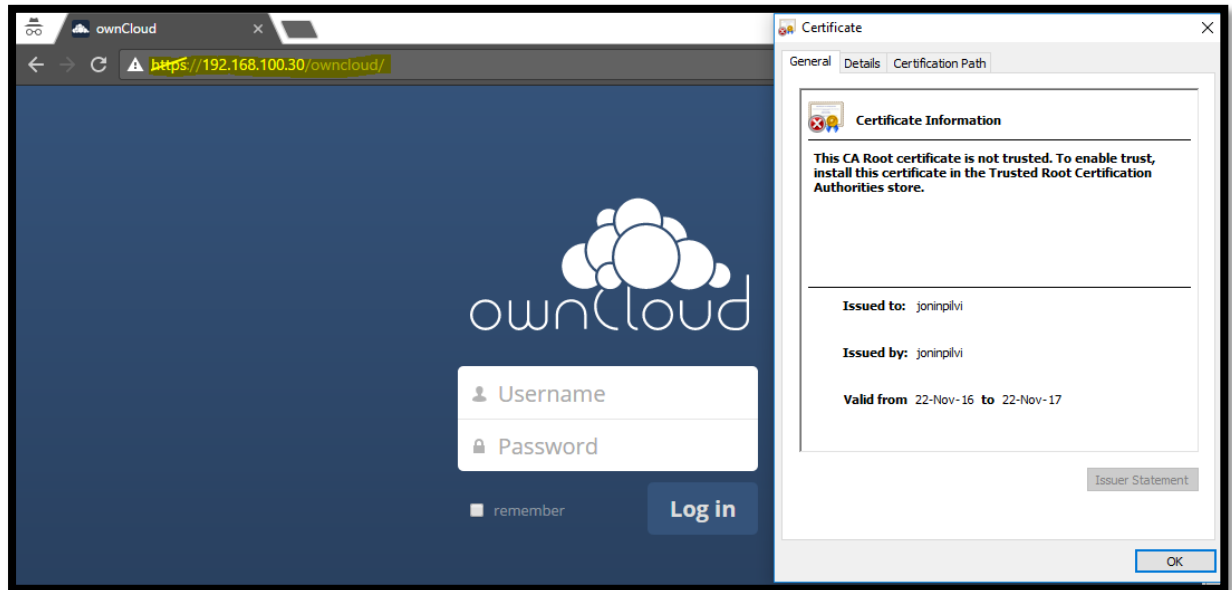


Kuvio 38. Admin sivun Cron asetukset.

6 Toimivuuden testaus

6.1 HTTPS uudelleenohjaus

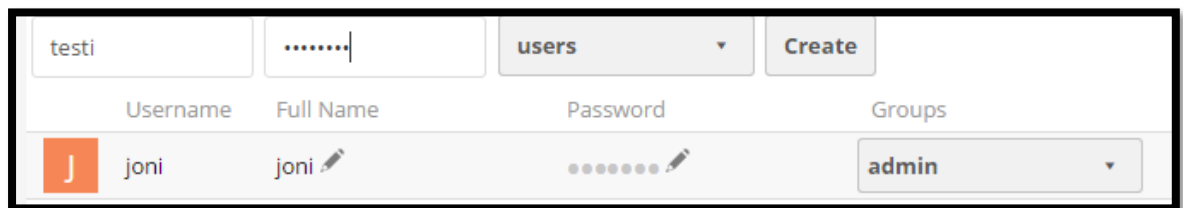
Kun menin selaimella palvelimeni sivuille, se uudelleenohjaa http protokollan https:si ja huomauttaa, että sertifikaatti ei ole luotettu sertifikaatti.



Kuvio 39. HTTPS uudelleenohjaus ja sertifikaatti

6.2 Tiedostojen vienti palvelimelle ja jakaminen

Luon ensin uuden testikäyttäjän Owncloudin admin – sivuilla.

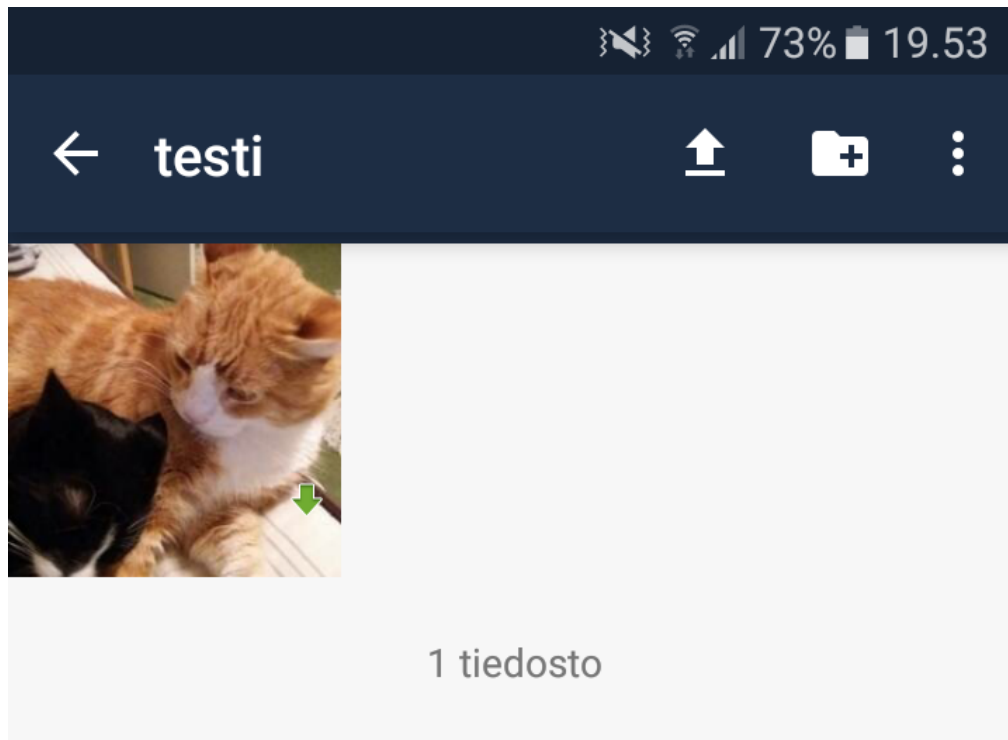


Kuvio 40. Testikäyttäjän luonti.

Sitten latsin puhelimelleni Owncloud sovelluksen Google Play – sovelluskaupasta.

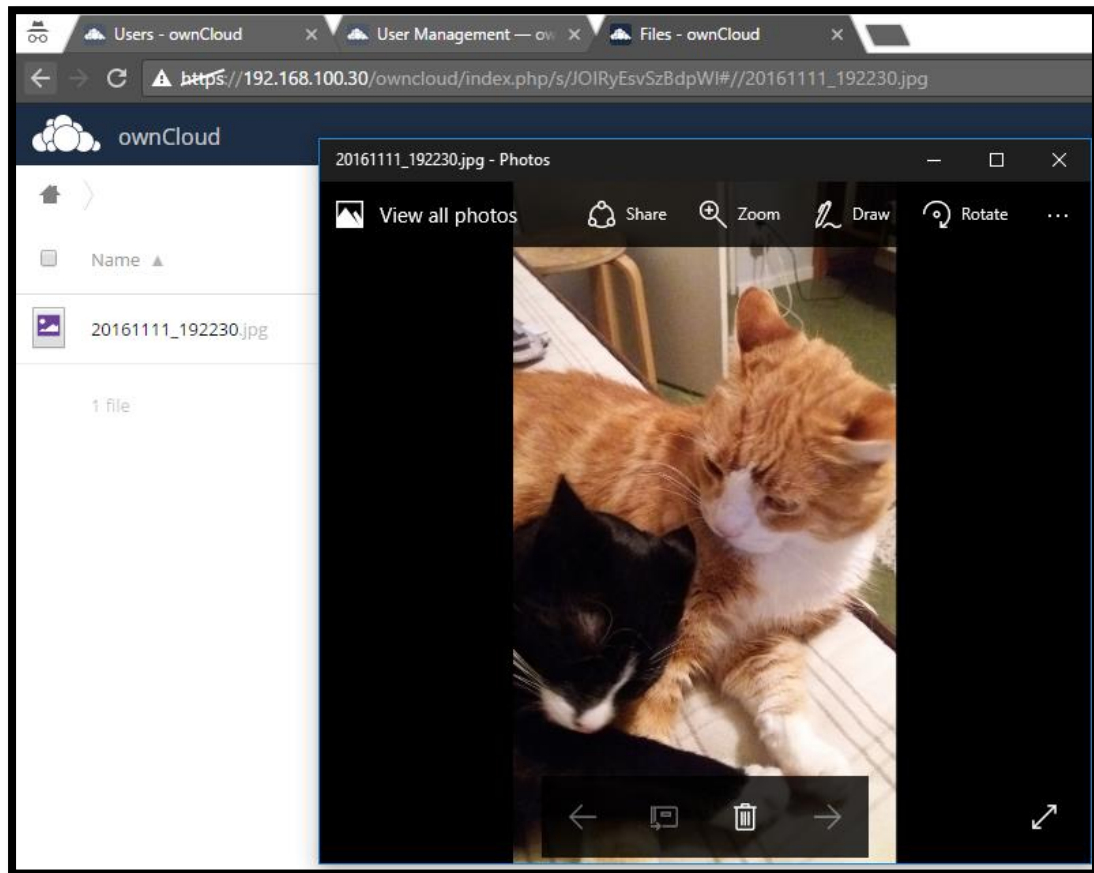
Kirjauduin puhelimellani sisään testikäyttäjänä ja loin uuden testikansion. Vein

palvelimelle uuden kuvan kännykästäni.



Kuvio 41. Näkymä kännykälläni, kun vein palvelimelle uuden tiedoston.

Jaoin linkin kuvaan, kirjoitin sen tietokoneeni selaimeen ja latasin kuvan. Kuva latautui onnistuneesti, eikä kuvien esikatselu toimi, koska se on config.php:ssä määriteltä pois päältä.



Kuvio 42. Kuvan lataaminen jaetusta linkistä

7 Kehitettävää tulevaisuudessa

7.1 Käyttö sisäverkon ulkopuolelta

En asettanut palvelimelle käyttöä sisäverkon ulkopuolelta, koska kaapelimodeemini Elisan Kotiboksi ei ole jostain syystä laittanut port forwarding – asetusten muokkaamista reitittimen sivuille. Huomasin tämän puutteen liian myöhään, enkä kerennyt kysymään Elisan asiakaspalvelusta, että onko porttien avaus mahdollista.

7.2 SELinux

SELinuxin asentamisessa tuli vastaan ongelmia. Debianin Jessie julkaisussa oli SELinuxin default policy tiedostoissa kriittisiä bugeja, joten se poistettiin Jessien virallisesta stable-repositoriosta. Kokeilin redditistä löydettyä ratkaisua, jossa rakennetaan policy itse kloonamalla tiedostot Tresys Technologyn GitHubista. Kloonasin GitHubista tiedostot kansioon, muokkasin retpolicy kansion sisältävä löytyvää build.conf:ia ja vaihdoin sinne käyttöjärjestelmän tiedot oikeiksi. Muokkasin /etc/selinux/conf tiedostoa ja asetin GitHubista kloonatun retpolicyn SELinuxin policyn lähteeksi.

SELinuxin asentaminen tyssäsi vaiheeseen, jossa tarvitsi rakentaa retpolicystä toimiva paketti make-työkalulla. Make load komento lopetti toimintansa alla olevaan virheeseen, johon en löytänyt netistä ratkaisua.

```
root@Debian:/etc/selinux/retpolicy# make conf && make && make install && make install-headers && make load
m4 -D self_contained_policy -D enable_mcs -D distro_debian -D init_systemd -D enable_ubac -D mls_num_sens=1
6 -D mls_num_cats=1024 -D mcs_num_cats=1024 -D hide_broken_symptoms support/divert.m4 policy/modules/kernel
/corenetwork.te.m4 support/undivert.m4 policy/modules/kernel/corenetwork.te.in \
```

```
make: execvp: /usr/sbin/semodule: Permission denied
Rules.modular:56: recipe for target 'load' failed
make: *** [load] Error 127
```

Kuviot 43-44. Make komennot ja make load virheilmoitus.

Ongelman olisi voinut välttää valitsemalla toisen käyttöjärjestelmän. SELinux policy on ollut poistettuna Jessien repositoriosta jo yli vuoden, joten en usko, että ratkaisu olisi tulossa lähiaikoina.