

## T-O4A, Linux-harjoitus 2

---

Tässä harjoituksessa asennetaan Apache, konfiguroidaan sitä, luodaan salasanasuojattu web-sivusto ja testataan php:n toiminta.

### Hieman linkkejä:

<http://fi.php.net/>, PHP-sivuston suomalainen peili.

<http://www.ohjelmointiputka.net/oppaat.php>, Ohjelmointiputka. Nippu ohjeita ja oppaita.

Hieman epätasalaatuisia, mutta kyllä näistä jotain selviää...

### Tehtäviä

1. Tarkista, mikä on Apache www-serverin tila komennolla

`service httpd status`

2. Jos palvelu ei ole päällä, laita se käyntiin:

`service httpd start`

3. Mitä tekee komento

`chkconfig --list` (huomaa, optiossa on kaksi tavuviivaa!)

Mitä `httpd`-rivillä lukee?

4. Tutustu `chkconfig`-komennon manuaalisivuihin ja aseta sen avulla `httpd`-demoni päälle. Miten tämä muuten eroaa `service`-komennosta?
5. Testaa `http`-serverin toiminta kokeilemalla selaimella surffata osoitteeseen `http://ip.os.oi.te`, jossa `ip.os.oi.te` on koneesi `ip`-osoite.
6. Kirjautu sisään jonakin tavallisena käyttäjänä. Tee tälle käyttäjälle kotihakemistoon hakemisto `public_html`. Tee tähän hakemistoon yksinkertainen `www`-sivu ja anna sille nimeksi `index.html`. Kokeile sen jälkeen jollakin selaimella `http://ip.os.oi.te/~ktunnus/` (`ip.os.oi.te` on koneesi `ip`-osoite ja `ktunnus` käyttäjän käyttäjätunnus) ja katso, tapahtuiko mitään.
7. Jos et päässyt hakemistoon, sait varmaan virheilmoituksia. Selvitä näiden perusteella, mistä asia on kiinni. Joudut myös tutkimaan Apachen logitiedostoja `/var/log/httpd` -kansioista. Tee sitten tarvittavia muutoksia Apachen konfigurointitiedostoon, joka on yleensä `/etc/httpd/conf/httpd.conf`. Tiedosto on tekstitiedosto ja sitä voi editoida millä editorilla haluaa.
8. Tee kotihakemistoosi `public_html`-niminen kansio ja sen alle toinen kansio, vaikkapa `testi`. Tee `testi`-kansioon jokin yksinkertainen `html`-sivu.

9. Testaa, että näet sivusi selaimella osoitteesta tyyliin `http://172.16.1.x/~ktunnus/testi/sivu.html` , jossa x on koneesi ip-osoitteen viimeinen numero ja ktunnus käyttäjätunnukseksi. Siirry eteenpäin tästä vasta sitten, kun sivu varmasti näkyy.
10. Mene tekemääsi hakemistoon ja kirjoita sinne jollakin editorilla tiedosto, jonka nimeksi annat `.htaccess`. (nimi alkaa siis pisteellä). Tähän tiedostoon kirjoitat seuraavanlaisen sisällön:

```
AuthUserFile /home/ktunnus/.htpasswd
AuthGroupFile /dev/null
AuthName "Anna salasana!"
AuthType Basic

<Limit GET POST PUT>
require user juuseri
</Limit>
```

Juuseri on esmerkissä käyttäjätunnus, jolla tulee pääsemään tuohon hakemistoon. Nyt pitää vielä luoda käyttäjän kotihakemistoon `.htpasswd` -tiedosto ja käyttäjätunnus ja sille salasana:

```
htpasswd -c .htpasswd juuseri
```

Huomaa `-c` -optio. Sitä tarvitaan vain ensimmäisellä kerralla kun tiedosto luodaan uusia käyttäjiä lisätään kantaan komentamalla

```
htpasswd .htpasswd toinen
```

Testaa nyt, toimiiko. Jos ei toimi, pitää käydä hieman virittämässä `httpd.conf`-tiedostoa.

# PHP

PHP on ohjelmointikieli, jolla voi rakentaa dynaamisia verkkosivuja. Yhdessä tietokantapalvelinten kuten PostgreSQL tai MySQL kanssa sillä voi liittää tietokantoja verkkosivuille.

1. Tarkista, oletko asentanut PHP:n. Jos olet, haku

```
rpm -qa | grep php
```

kertoo sen. Jos et ole asentanut, asenna se komennolla

```
yum install php
```

Voit katsoa *yum list | grep php* -komennolla, mitä kaikkea php-alkuista on tarjolla. Tietokanta- ynnä muut liittymät ovat eri paketeissa.

2. Tee yksinkertainen php-ohjelma, jossa testaat php-tulkin toimintaa:

```
<html>
<head>
<title>php-testi</title>
</head>
<body>
<?php
echo "moi, toimii";
?>
</body></html>
```

3. Vaihda echoon "moi, toimii":n tilalle *phpinfo()* ja katso, mitä se tulostaa.
4. Tutustu alussa olleiden linkkien avulla siihen, miten PHP:llä toteutetaan tietokantaliittymä. Voit myös kokeilla, saatko tehdyksi yksinkertaisen liittymän.