



## Koventaminen – Labra 5

### Ryhmä 3

Juha-Matti Hietala

Markus Pollari

Topi Liljeqvist

Maija Virta

Oppimistehtävä

Huhtikuu 2023

Tekniikan ala

Tieto- ja viestintätekniikan tutkinto-ohjelma (AMK)

## Sisältö

<b>1</b>	<b>Johdanto .....</b>	<b>2</b>
<b>2</b>	<b>Teoria .....</b>	<b>2</b>
2.1	MFA (Multi Factor Authentication) .....	2
<b>3</b>	<b>Dokumentointi - 2FA Wordpressiin ja Centosiin .....</b>	<b>2</b>
3.1	2FA lisäosan lisäys Wordpressiin.....	2
3.2	2FA Centosiin.....	7
<b>4</b>	<b>Dokumentointi - SSH Daemon käyttämään Google Authenticatoria .....</b>	<b>10</b>
4.1	Sshd_config .....	10
4.2	SSH PAM säännöt .....	12
<b>5</b>	<b>Pohdinta.....</b>	<b>13</b>
	<b>Lähteet .....</b>	<b>14</b>

## Kuvat

Kuva 1.	"Let us help you get started" wizard.....	3
Kuva 2.	2FA methods .....	4
Kuva 3.	2FA kaikille käyttäjille .....	4
Kuva 4.	2FA konfiguroidaan heti.....	5
Kuva 5.	QR-koodi.....	5
Kuva 6.	Vahvistuskoodin validointi .....	6
Kuva 7.	Wordpress pyytää vahvistuskoodia .....	6
Kuva 8.	Kirjautuminen onnistui.....	7
Kuva 9.	EPEL asennus (oli jo asennettuna) .....	7
Kuva 10.	Google Authenticatorin asennus .....	8
Kuva 11.	QR-koodi, salasavain sekä pyyntö kirjoittaa koodi sovelluksesta.....	9
Kuva 12.	Emergency scratch codes .....	9
Kuva 13.	Kysymyksiä .....	10
Kuva 14.	UsePAM .....	11
Kuva 15.	ChallengeResponseAuthentication .....	11
Kuva 16.	Google Authenticator käyttöön .....	12
Kuva 17.	SSH-kirjautuminen vaatii Google Authenticatorin .....	12

# 1 Johdanto

Dokumentaatio on osana koventamisen kurssin (TTC6050-3002) laboratorioharjoituksia. Lab5 tarkoituksena on tutustua ja ottaa käyttöön MFA (Multi Factor Authentication) wordpressiin ja www-palvelimen SSH-kirjautumiseen.

Labran teorialla ja harjoituksella ryhmän jäsenet saavat taidot ja ymmärryksen, miten MFA otetaan käyttöön.

Harjoituksen kokonaisuus dokumentoidaan kuvankaappauksilla, joiden avulla havainnollistetaan harjoitustehtävät. Lisäksi harjoitustyön alussa esitetään teoria Multi Factor Authenticationista sekä lopussa pohdinta harjoitustyön tekemisestä.

## 2 Teoria

### 2.1 MFA (Multi Factor Authentication)

MFA (Multi Factor Authentication) tarkoittaa monivaiheista tunnistautumista. Se on turvallisuusmekanismi, joka käyttää useampaa kuin yhtä tunnistustekijää, kuten salasanaa, käyttäjänimeä ja puhelinsovellusta. Tämä tekee kirjautumisesta turvallisempaa, sillä hyökkääjän on vaikeampi päästä käsiksi tilille, vaikka hänellä olisi käyttäjän tiedot. Yleisimpiä MFA:n käyttämiä tunnistustekijöitä ovat älypuhelimeen lähetettävä tekstiviesti, mobiilisovellus, joka generoi yhden käyttökerran salasanan tai biometriset tunnistetiedot, kuten sormenjälki tai kasvojentunnistus. (What is: Multifactor Authentication. N.d.)

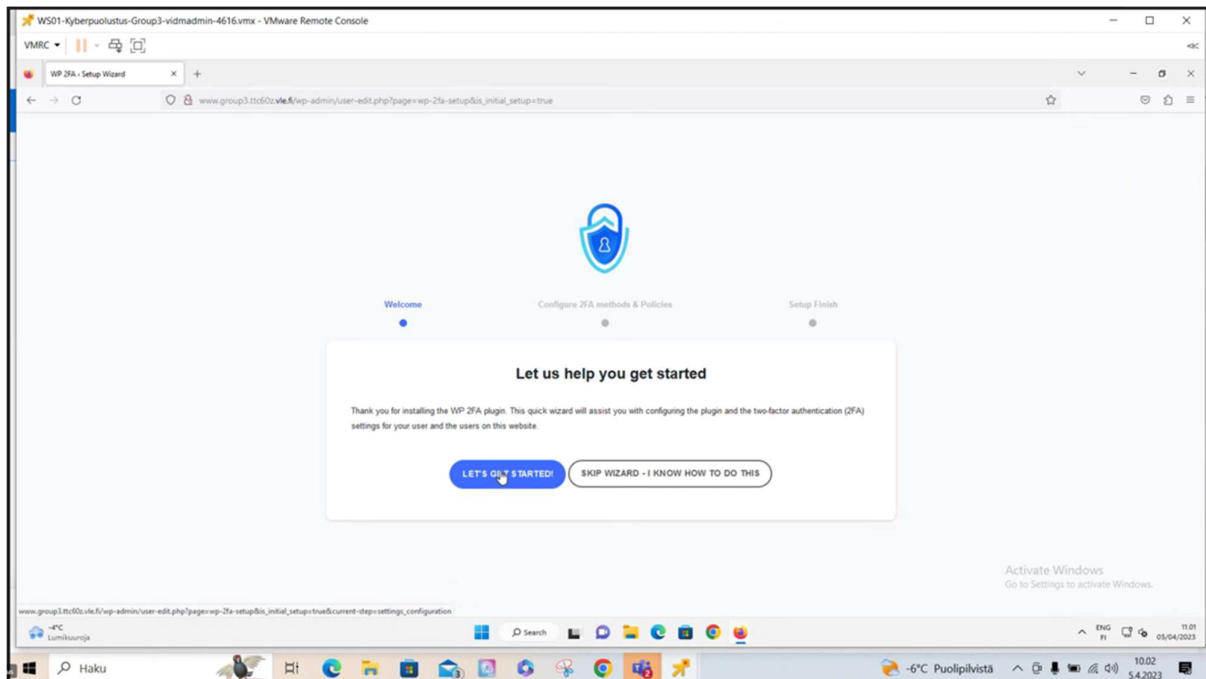
## 3 Dokumentointi - 2FA Wordpressiin ja Centosiin

### 3.1 2FA lisäosan lisäys Wordpressiin

Etsittiin Wordpressin admin-paneelin "plugins" osiosta lisäosa nimeltä "WP 2fa" ja asennettiin.

Aikaisemman labran ongelmanratkonnassa olimme jo lisänneet oikeuksia WWW-serverille, joten ohjeissa näkyvää virheilmoitusta "could not create directory" ei meillä ilmaantunut, vaan pystyimme suoraan asentamaan lisäosan.

Lisäosan asennuttua ja aktivoitua, ilmaantui "Let us help you get started" wizard, jossa aloitimme asennuksen painamalla "Let's get started!" nappia. Esitetty kuvassa 1.



Kuva 1. "Let us help you get started" wizard

Seuraavassa vaiheessa otettiin "One-time code via email (HOTP) pois ja varmistettiin, että "One-time code via 2FA App (TOTP)" on aktiivisena. Esitetty kuvassa 2.

1 2FA methods 2 Alternative methods 3 2FA policy 4 Grace period

### Which 2FA methods can your users use?

When you uncheck any of the below 2FA methods it won't be available for your users to use. You can always change this later on from the plugin's settings.

☒ One-time code via 2FA App (TOTP)

When using this method, users will need to configure a 2FA app to get the one-time login code. The plugin supports all standard 2FA apps. Refer to the [guide on how to set up 2FA apps](#) for more information. Allowing users to set up a secondary 2FA method is highly recommended. You can do this in the next step of the wizard. This will allow users to log in using an alternative method should they, for example lose access to their phone.

☐ One-time code via email (HOTP) - ensure email deliverability with the free plugin [WP Mail SMTP](#).

When using this method, users will receive the one-time login code over email. Therefore, email deliverability is very important. Users using this method should whitelist the address from which the codes are sent. By default, this is the email address configured in your WordPress. You can run an email test from the plugin's settings to confirm email deliverability. If you have had email deliverability / reliability issues, we highly recommend you to install the free plugin [WP Mail SMTP](#).

Allowing users to set up a secondary 2FA method is highly recommended. You can do this in the next step of the wizard. This will allow users to log in using an alternative method should they, for example lose access to their phone.

CONTINUE SETUP

Kuva 2. 2FA methods

Valittiin vaihtoehtoista, että 2FA laitetaan aktiiviseksi kaikille käyttäjille. Esitetty kuvassa 3.

Welcome Configure 2FA methods & Policies Setup Finish

1 2FA methods 2 Alternative methods 3 2FA policy 4 Exclude users 5 Grace period

### Do you want to enforce 2FA for some, or all the users?

When you enforce 2FA the users will be prompted to configure 2FA the next time they login. Users have a grace period for configuring 2FA. You can configure the grace period and also exclude user(s) or role(s) in this settings page. [Learn more.](#)

☒ All users

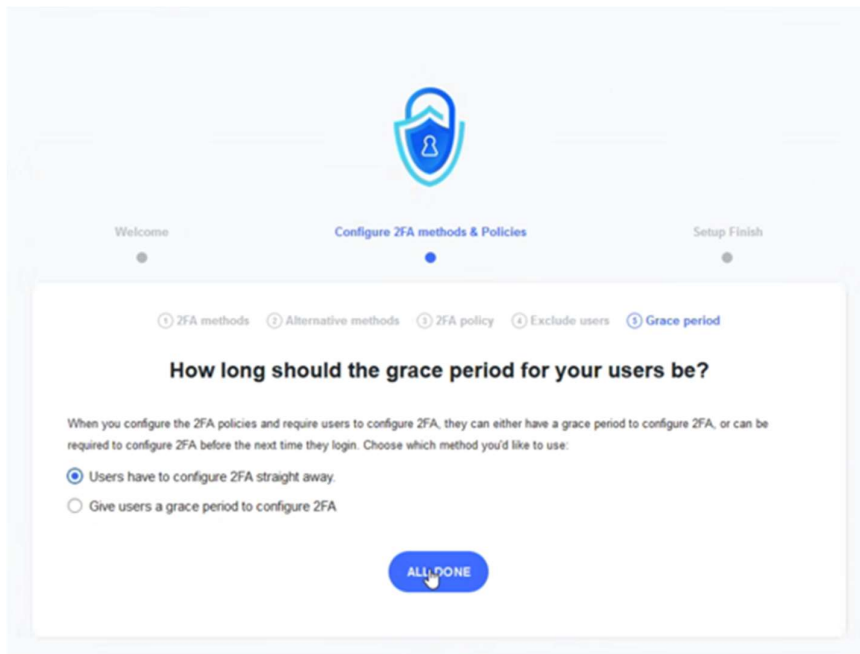
☐ Only for specific users and roles

☐ Do not enforce on any users

CONTINUE SETUP

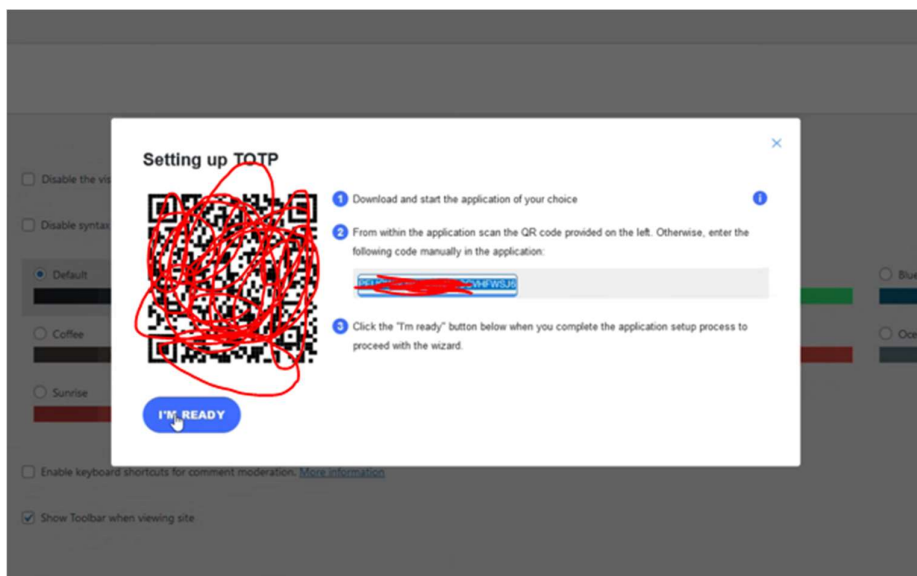
Kuva 3. 2FA kaikille käyttäjille

Käyttäjien täytyy konfiguroida 2FA heti. Esitetty kuvassa 4.



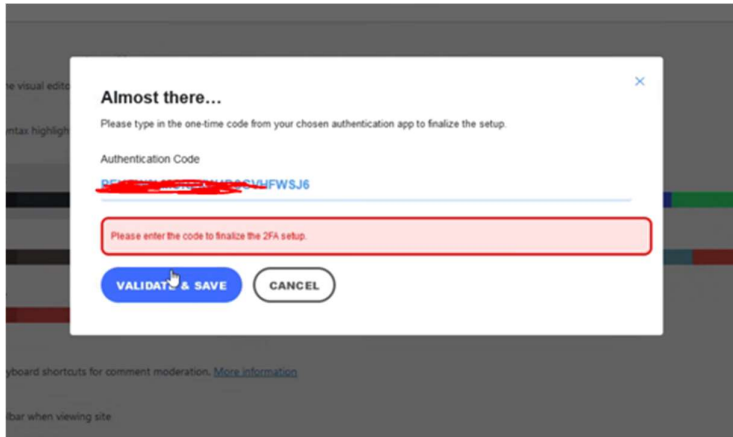
Kuva 4. 2FA konfiguroidaan heti

Seuraavaksi ruudulle ilmestyi QR-koodi ja vahvistuskoodi. Mikäli jostain syystä ei saa skannattua QR-koodia valittuun TOTP-authenticator ohjelmaan, pystyy sen lisäämään manuaalisesti vahvistuskoodilla. Meillä QR-skannaus toimi ongelmitta Google Authenticatorissa. Esitetty kuvassa 5.

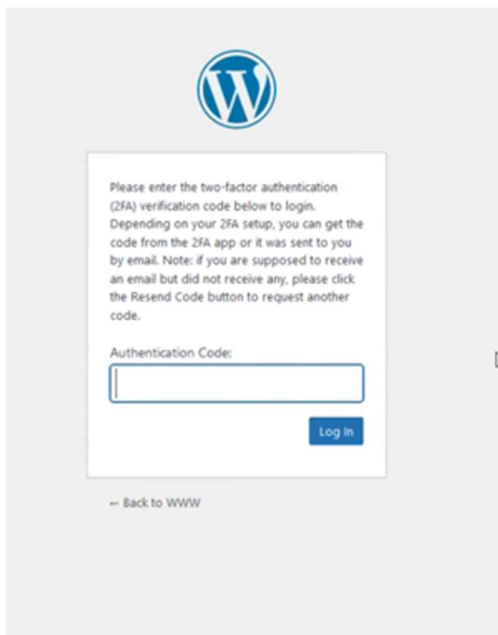


Kuva 5. QR-koodi

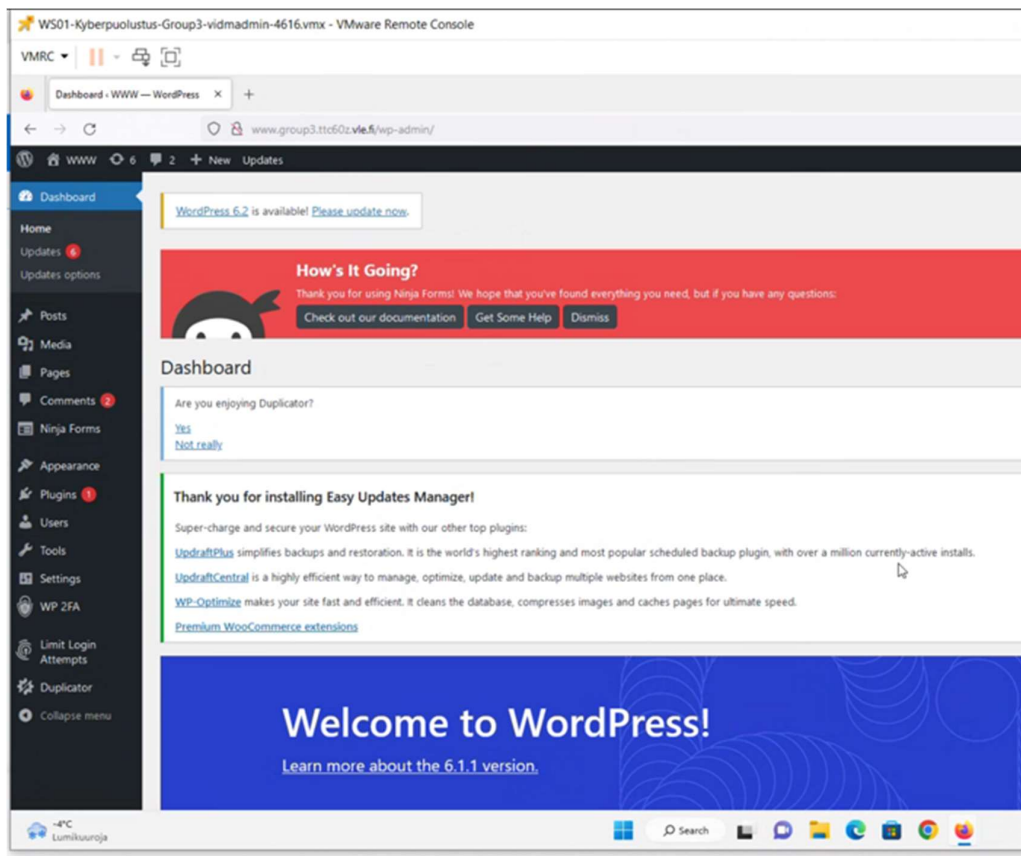
Asennus kysyi vielä lopuksi edellisessä vaiheessa annettua vahvistuskoodia ja painettuamme "Validate & Save", pyysi Wordpress vahvistuskoodia yrittäessämme kirjautua uudestaan. Kirjoitettuamme Google Authenticatorissa näkyvän koodin, kirjautuminen onnistui. Esitetty kuvissa 6-8.



Kuva 6. Vahvistuskoodin validointi



Kuva 7. Wordpress pyytää vahvistuskoodia



Kuva 8. Kirjautuminen onnistui

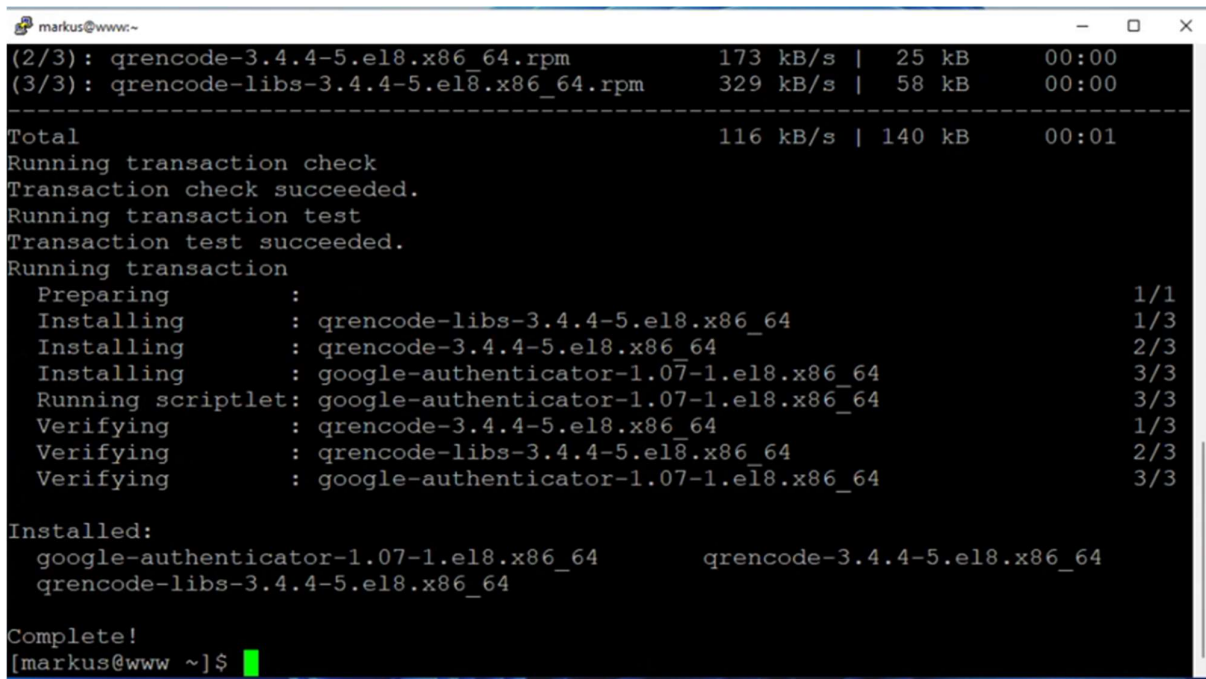
## 3.2 2FA Centosiin

Lisätään 2FA WWW-serverille SSH-kirjautumisiin. Salausavain ja QR-koodi luotiin jokaiselle käyttäjälle erikseen, esitetty miten lisätään yhdelle. Ensiksi asennettiin Google Authenticator EPEL (Extra Package for Enterprise Linux) reposta komennoilla "sudo dnf install -y epel-release" ja "sudo dnf install -y google-authenticator qrencode qrencode-libs". Esitetty kuvissa 9 ja 10.

```
[markus@www ~]$ sudo dnf install -y epel-release
[sudo] password for markus:
Sorry, try again.
[sudo] password for markus:
Last metadata expiration check: 2:00:54 ago on Wed 05 Apr 2023 08:13:38 AM EEST.
Package epel-release-8-18.el8.noarch is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[markus@www ~]$
```

Kuva 9. EPEL asennus (oli jo asennettuna)

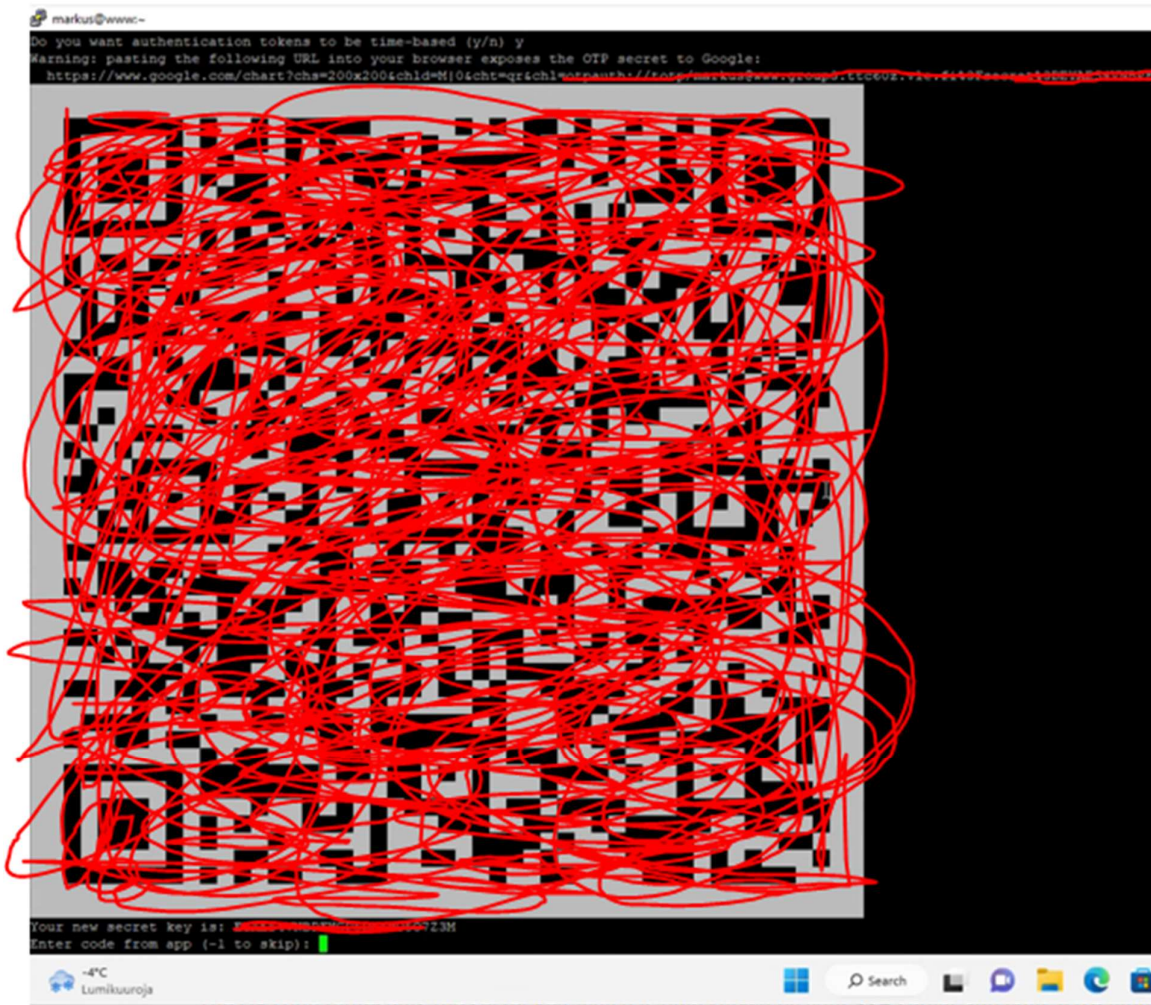




```
markus@www:~  
(2/3): gencode-3.4.4-5.el8.x86_64.rpm      173 kB/s | 25 kB    00:00  
(3/3): gencode-libs-3.4.4-5.el8.x86_64.rpm 329 kB/s | 58 kB    00:00  
-----  
Total                                116 kB/s | 140 kB    00:01  
Running transaction check  
Transaction check succeeded.  
Running transaction test  
Transaction test succeeded.  
Running transaction  
  Preparing      :                                1/1  
  Installing    : gencode-libs-3.4.4-5.el8.x86_64 1/3  
  Installing    : gencode-3.4.4-5.el8.x86_64      2/3  
  Installing    : google-authenticator-1.07-1.el8.x86_64 3/3  
  Running scriptlet: google-authenticator-1.07-1.el8.x86_64 3/3  
  Verifying     : gencode-3.4.4-5.el8.x86_64      1/3  
  Verifying     : gencode-libs-3.4.4-5.el8.x86_64 2/3  
  Verifying     : google-authenticator-1.07-1.el8.x86_64 3/3  
  
Installed:  
  google-authenticator-1.07-1.el8.x86_64      gencode-3.4.4-5.el8.x86_64  
  gencode-libs-3.4.4-5.el8.x86_64  
  
Complete!  
[markus@www ~]$
```

Kuva 10. Google Authenticatorin asennus

Asennuksen jälkeen ajettiin komento "google-authenticator -s ~/.ssh/google\_authenticator" luodaksemme uuden salausavaimen käyttäjän ~/.ssh kansioon. Kysymykseen "Do you want authentication tokens to be time-based?" valittiin "yes", jonka jälkeen ikkunaan ilmestyy QR-koodi, salausavain sekä pyyntö kirjoittaa QR-koodin skannauksen jälkeen Google Authenticator -sovelluksessa näkyvä koodi. Esitetty kuvassa 11.



Kuva 11. QR-koodi, salausavain sekä pyyntö kirjoittaa koodi sovelluksesta

Kirjoitettuaamme koodin sovelluksesta, tulostui ruudulle käyttäjäkohtaiset "emergency scratch codes", jotka tuli ottaa talteen mahdollisia ongelmatilanteita varten. Esitetty kuvassa 12.



Kuva 12. Emergency scratch codes

Seuraaviin kysymyksiin vastattiin kaikkiin "yes". Esitetty kuvassa 13.

```
Do you want me to update your "/home/markus/.ssh/google_authenticator" file? (y/n) y

Do you want to disallow multiple uses of the same authentication
token? This restricts you to one login about every 30s, but it increases
your chances to notice or even prevent man-in-the-middle attacks (y/n) y

By default, a new token is generated every 30 seconds by the mobile app.
In order to compensate for possible time-skew between the client and the server,
we allow an extra token before and after the current time. This allows for a
time skew of up to 30 seconds between authentication server and client. If you
experience problems with poor time synchronization, you can increase the window
from its default size of 3 permitted codes (one previous code, the current
code, the next code) to 17 permitted codes (the 8 previous codes, the current
code, and the 8 next codes). This will permit for a time skew of up to 4 minutes
between client and server.
Do you want to do so? (y/n) y

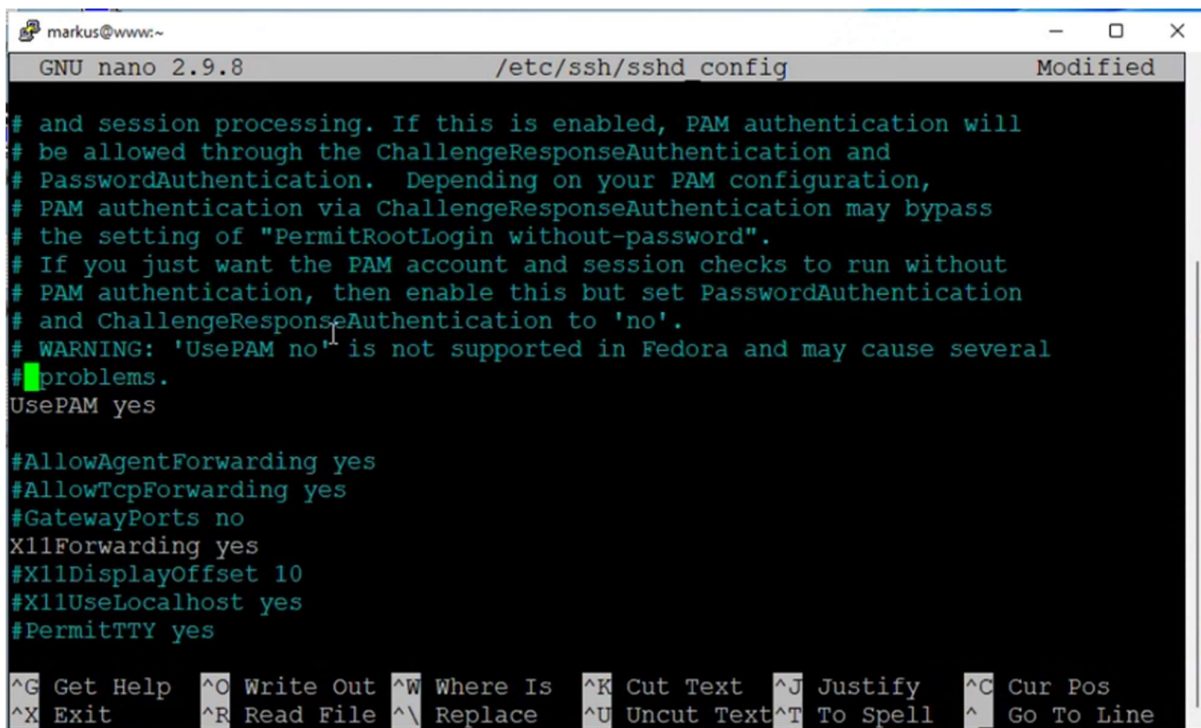
If the computer that you are logging into isn't hardened against brute-force
login attempts, you can enable rate-limiting for the authentication module.
By default, this limits attackers to no more than 3 login attempts every 30s.
Do you want to enable rate-limiting? (y/n) y
[markus@www ~]$
```

Kuva 13. Kysymyksiä

## 4 Dokumentointi - SSH Daemon käyttämään Google Authenticatoria

### 4.1 Sshd\_config

Avattiin SSH server konfigurointi tiedosto komennolla "sudo nano /etc/ssh/sshd\_config" ja muokattiin rivit "UsePAM" sekä "ChallengeResponseAuthentication" arvoin "yes" ja tallennettiin tiedosto. Esitetty kuvissa 14 ja 15.



```

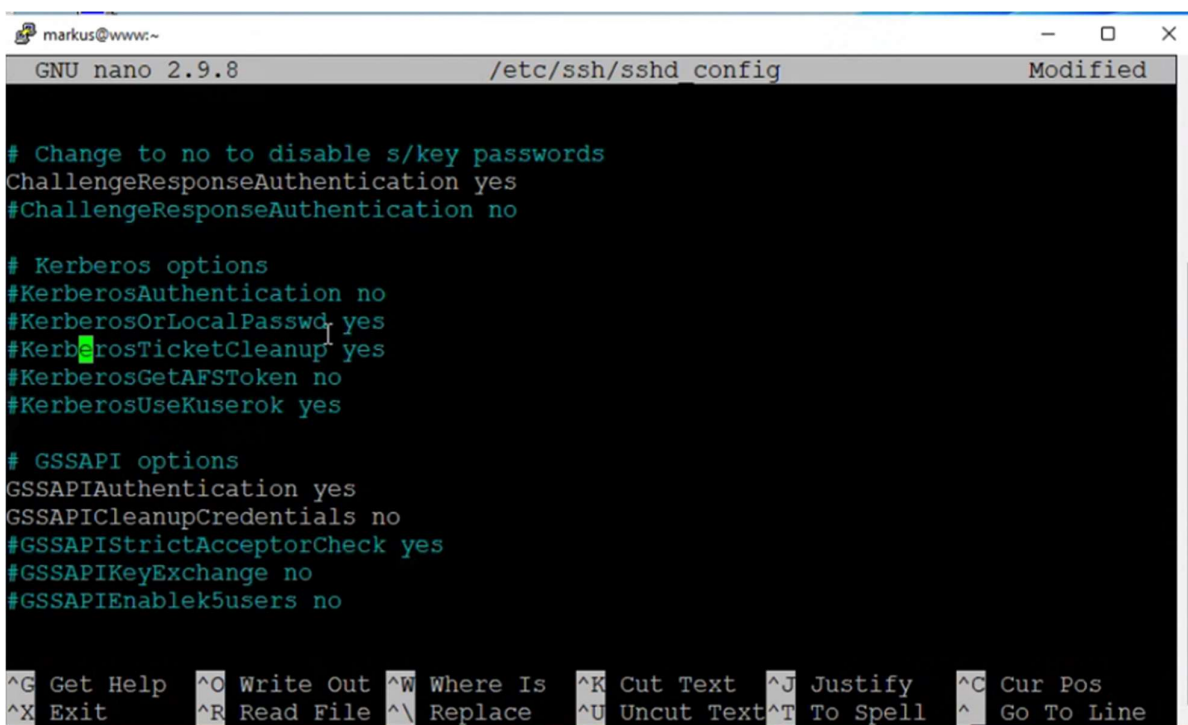
markus@www:~
GNU nano 2.9.8 /etc/ssh/sshd_config Modified
# and session processing. If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
# WARNING: 'UsePAM no' is not supported in Fedora and may cause several
# problems.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line

```

Kuva 14. UsePAM



```

markus@www:~
GNU nano 2.9.8 /etc/ssh/sshd_config Modified

# Change to no to disable s/key passwords
ChallengeResponseAuthentication yes
#ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no
#KerberosUseKuserok yes

# GSSAPI options
GSSAPIAuthentication yes
GSSAPICleanupCredentials no
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no
#GSSAPIEnablek5users no

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line

```

Kuva 15. ChallengeResponseAuthentication



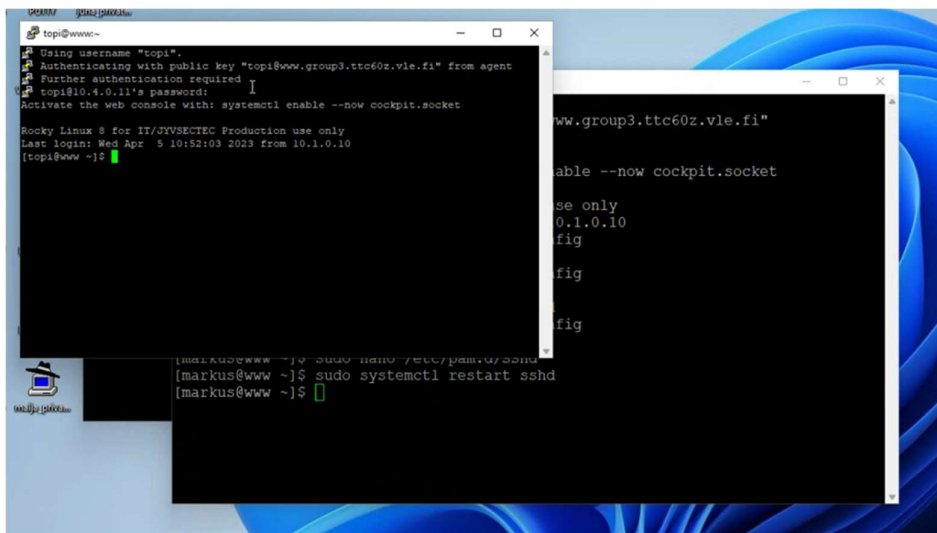
## 4.2 SSH PAM säännöt

Avattiin tiedosto SSH daemonin PAM (Pluggable Authentication Module) sääntöihin komennolla "sudo nano /etc/pam.d/ssh". Lisättiin tiedostoon sääntö "auth required pam\_google\_authenticator.so secret=\${HOME}/.ssh/google\_authenticator" ja tallennettiin. Esitetty kuvassa 16.

```
account    required    pam_sepermit.so
account    required    pam_nologin.so
account    include     password-auth
password    include     password-auth
# pam_selinux.so close should be the first session rule
session     required    pam_selinux.so close
session     required    pam_loginuid.so
# pam_selinux.so open should only be followed by sessions to be executed in the$
session     required    pam_selinux.so open env_params
session     required    pam_namespace.so
session     optional    pam_keyinit.so force revoke
session     optional    pam_motd.so
session     include     password-auth
session     include     postlogin
#two-factor authentication via Google Authenticator
auth        required    pam_google_authenticator.so secret=${HOME}/.ssh/google_au$
```

Kuva 16. Google Authenticator käyttöön

Jotta muutokset tulivat voimaan, käynnistettiin SSH daemon uudelleen komennolla "sudo systemctl restart sshd". Testattiin asetuksia ottamalla uusi SSH-yhteys WWW-serverille. Yhteyttä muodostaessa serveri ilmoittaa "Further authentication required" ja pyytää käyttäjän salasanaa. Kirjoitettuaamme koodin Google Authenticatorista, kirjautuminen onnistui. Esitetty kuvassa 17.



Kuva 17. SSH-kirjautuminen vaatii Google Authenticatorin

## 5 Pohdinta

Dokumentaatio oli osana koventamisen kurssin (TTC6050-3002) laboratorioharjoituksia. Lab5 tarkoituksena oli tutustuttaa ryhmän jäsenet MFA (Multi Factor Authentication) käyttöönottoon wordpressiin sekä www-palvelimen SSH-kirjautumiseen.

Harjoitustyö aloitettiin 2FA lisäosan lisäämisellä Wordpressiin, jonka tekeminen ohjeiden mukaan meni ongelmitta. Siirryimme lisäämään 2FA WWW-serverille SSH-kirjautumisiin ja tässäkään ei tullut ongelmaa ja lopuksi vielä testasimme kirjautumisen jokaisen tunnuksella.

Harjoitustyö ajallisesti oli nopea toteuttaa ja ohjeet olivat selkeät. Eli tällä kertaa suurempia ongelmia ei ilmennyt ja jokainen ryhmän jäsen tiesi jo entuudestaan 2FA käytöstä, mutta sen käyttöönotto oli uutta. Mielenkiintoinen labra ja olisimme todennäköisesti päätyneet tekemään tämän jo aikaisemmin osana kovennuksia, jos opettaja ei olisi etukäteen huomauttanut, että aiheesta tulee oma labransa.

## Lähteet

What is: Multifactor Authentication. N.d. Microsoft verkkosivut. Viitattu 18.4.2023. <https://support.microsoft.com/en-us/topic/what-is-multifactor-authentication-e5e39437-121c-be60-d123-eda06bddf661>

