

# **TunneliClient käyttöohje**

# 1 TUNNELICLIENT

## 1.1 Yleistä dokumentista

Tämän dokumentin tarkoituksena on kertoa, miten ohjelmistokehittäjä saa käyttöönsä TunneliClient asiakassovelluksen ja antaa yleiskuva sen toiminnasta.

## 1.2 Yleiskuvaus

TunneliClient on Java-kielellä toteutettu sovellus, joka hyödyntää Ilmoitin.fi-palvelun Tunneli-rajapintaa. Sovelluksen avulla voi lähettää tulo- ja kausiveroilmoituksia Verohallinnon käsittelyyn.

# 2 ASENNUSOHJE

TunneliClientin jaetavassa paketissa (tunnelicl-<versio>-exe.jar) on kaikki sovelluksen suorittamiseen tarvittavat tiedostot. Koko paketin voi asentaa haluamaansa hakemistoon, esim. C:/TunneliCl.

## 2.1 Sovelluksen ympäristö

TunneliClient tarvitsee suoritusta varten Java-virtuaalikoneen. Koska kyseessä on täysimittainen ohjelma, joka mm. lukee ja kirjoittaa tiedostoja, sen suorittamiseen eivät riitä sovelmia varten kehitetyt Web-selainten suojatut Java-ympäristöt. Koneessa, jossa tarkistusmoduulia halutaan käyttää, on oltava

**Java 2 Runtime Environment, Standard Edition (versio 1.5 tai uudempi),**

joka on maksutta saatavissa suurimmalle osalle yleisiä tietokoneita ja käyttöjärjestelmiä Sunin verkkosivuilta (<http://java.sun.com/downloads/>). Sivuilta löytyvät myös ympäristön asennusohjeet.

# 3 KÄYTÖNOHJE

Sovellusta voidaan kutsua joko toisen ohjelman sisältä tai suorittaa itsenäisenä ohjelmana.

## 3.1 Ajovaihtoehdot

### 3.1.1 Kutsuminen toisesta sovelluksesta

Kutsuttaessa sovellusta, annetaan parametrina ilmoitusdata, tarkastustuloksen kieli, Tunneli-palvelun käyttäjätunnus ja salasana.

Esimerkki sovelluksen kutsusta toisesta java-kielisestä ohjelmasta:

```
TunneliCl palvelu = new TunneliCl(TunneliKIProp.getAjoymparisto());  
palvelu.laheta(ilmoitusdata, TunneliKIProp.getKieli(), kayttajatunnus, salasana);
```

Parametri **ilmoitusdata** sisältää Verohallintoon lähetettävän aineiston. Parametri ei voi olla tyhjä.

Parametri **kieli** kertoo, millä kielellä tarkistuksen tulokset halutaan saada. Vaihtoehtoina on suomen (FI), ruotsin (SE) ja englannin kieli (EN). Kieli merkitään parametrin arvoksi suluissa nähtävällä tavalla eikä se voi olla tyhjä.

Parametrin **kayttajatunnus** on Tunneli-palvelun käyttäjätunnus.

Parametri **salasana** on Tunneli-palvelun salasana.

*Lähetä(...)* -metodin kutsu palauttaa *TunneliTulos* tyyppisen olion, joka sisältää seuraavat metodit:

*getHttpStatusCode()* – palauttaa kutsun http-tilakoodin.

*getTunneliTulos()* – palauttaa Tunneli-palvelun vastauksen lähetettyyn ilmoitukseen. Vastauksen muoto käsitellään tarkemmin dokumentin *Ilmoitinfi\_tunnelipalvelun\_toimintoprosessikuvaus\_v1.2.pdf* kohdissa 4 ja 5.

*getVirheTeksti()* – palauttaa mahdollisen virhetekstin.

### 3.1.2 Ajaminen omana sovelluksenaan

1 = Ohjelmaa ajetaan esimerkissä samasta hakemistosta, missä on sovelluksen suoritettava tiedosto. Suoritettava tiedosto voi myös sijaita muualla kuin ajettavassa hakemistossa ja silloin ilmoitetaan koko polku.

2 = Tunnelipalvelun salasana

3 = Tunnelipalvelun käyttäjätunnus

4 = Ilmoitustiedosto sijaitsee esimerkissä ajettavassa hakemistossa. Ilmoitustiedosto voi sijaita myös vapaavalintaisessa hakemistossa, jolloin tiedoston polku ilmoitetaan parametrina.

java	-jar	tunnelicl-1.1-exe.jar	salasana	käyttäjätunnus	Ilmoitus.txt
		2	3	4	5

Sovellus tulostaa kutsun tuloksen näytölle. Lisäksi lähetetyt ja vastaanotetut sanomat tallennetaan tunnelicl.properties –tiedostossa määriteltyyn hakemistoon (kts. 3.2 Properties -tiedosto). Vastauksen muoto käsitellään tarkemmin dokumentin *Ilmoitinfi\_tunnelipalvelun\_toimintoprosessikuvaus\_v1.2.pdf* kohdissa 4 ja 5.

## 3.2 Properties -tiedosto

Haluttaessa TunneliClient tallentaa lähetetyt ja vastaanotetut sanomat sovelluksen ajohakemiston alla olevaan msg-hakemistoon. Tätä toiminnallisuutta hallitaan tunnelicl.properties –tiedostosta käsin laittamalla *tunneli.lokita.sanom*at –parametrin arvoksi **true**. Sanomat tallentuvat hakemistoon xml-muodossa.

Myös Tunneli-palvelun vastaussanomien http-headerit on mahdollista tallentaa. Laittamalla *tunneli.lokita.http.headerit* –parametrin arvoksi **true**, sovellus tallentaa headerit tekstitiedostoon, samaan hakemistoon sanomien kanssa.

*tunneli.sanomaloki.polku* –parametrin avulla voi muuttaa sanomien ja headeritiedostojen tallennuspaikkaa.