TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Rihard Rivis 222914IAIB

# eDelivery SML ja SMP kasutamine eFTI süsteemis

Bakalaureusetöö

Juhendaja: Ulrika Hurt
MSc

Kaasjuhendaja: Tarvo Treier
MSc

Tallinn 2026

## Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Rihard Rivis

10.05.2026

# Lühikokkuvõte

Euroopa liidu eFTI (Electronic Freight Transport Information) regulatsioon näeb ette veoinfo digitaalse vahetamise pädevate asutuste ja veoettevõtete vahel. Liikmesriigid peavad olema aastaks 2027 valmis vastu võtma veoinfot digitaalsel kujul ning muutma selle kättesaadavaks pädevatele asutustele. Sellest tulenevalt peab olema igal liikmesriigil oma eFTI värav, mis infot osapoolte vahel edastama hakkab.

Regulatsiooni kohalset on liikmesriikide ja riigisisese andmevahetuse aluseks eDelivery infovahetuse süsteem. Riikide vahel kasutatake staatilist eDelivery konfiguratsiooni, mille kohaselt seadistatakse vajaminevad sertifikaadid ning muud parameetrid manuaalselt. eFTI värav peab võimaldama veoettevõtetele ehk eFTI platvormidele ka eDelivery liidest, kuid võib pakkuda ka teisi lahendusi nagu näiteks REST liides.

Platvormidele loodud eDelivery lahendus peab kasutama dünaamilist konfiguratsiooni, mis automatiseerib sertifikaatide ning muude parameetrite leidmist. Dünaamiliselt metaandmete leidmiseks on eDeliverys kasutusel kaks lisakomponenti: SMP (Teenuse metaandmete avaldaja) ja SML (Teenuse metaandmete leidja).

Selle lõputöö eesmärk on uurida kuidas töötavad eDelivery dünaamilise otsingu komponendid ning luua prototüüplahendus. Oluline on analüüsida olemasolevaid lahendusi ning luua jätkusuutlik tarkvara, mis suudaks töödelda võimalikult palju päringuid.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 34 leheküljel, 4 peatükki, 12 joonist, 7 tabelit.

# Abstract

## Using eDelivery SML and SMP in eFTI system

[YOUR TEXT GOES HERE]

The thesis is in Estonian and contains 34 pages of text, 4 chapters, 12 figures, 7 tables.

# Lühendite ja mõistete sõnastik

| | |
|---|---|
| API | Rakendusliides (*Application Programming Interface*) |
| CPU | Keskseade (*Central Processing Unit*) |
| IDE | Integreeritud programmeerimiskeskkond (*Integrated Development Environment*) |
| IOT | Asjade Internet (*Internet Of Things*) |
| VM | Virtuaalmasin (*Virtual Machine*) |

# Sisukord


1 Sissejuhatus ... 11
1.1 eFTI taust ... 11
1.2 eDelivery taust ... 12
1.3 SMP ja SML taust ... 13
1.4 Probleem ... 14
1.5 Eesmärk ... 14
1.6 Metoodika ... 15
2 eDelivery valmiskomponendid ... 16
2.1 eDelivery juurdepääsupunkt (AP) ... 16
2.1.1 Domibus ... 17
2.1.2 Harmony ... 17
2.1.3 Phase4 ... 18
2.2 eDelivery teenuse metaandmete avaldaja (SMP) ... 19
2.2.1 Phoss SMP ... 19
2.2.2 Harmony SMP ... 19
2.3 eDelivery teenuse metaandmete leidja (SML) ... 20
2.3.1 DomiSML ... 20
3 Näidislahenduse loomine ... 21
3.1 Staatiline eDelivery konfiguratsioon ... 21
3.1.1 Juurdepääsupunktide konteinerite seadistamine ... 21
3.1.2 Juurdepääsupunkti seadistused ... 22
3.1.3 Digitaalsete sertifikaatide ja võtmete seadistus ... 23
3.1.4 PMode seadistus ... 24
3.1.5 WS plugin seadistus ... 24
3.1.6 WS plugina kasutamine ... 24
3.2 Dünaamiline eDelivery konfiguratsioon ... 26
3.2.1 domibus.properties seadistused ... 26

3.2.2 PMode seadistus ... 27
3.2.3 Sertifikaadid ... 27
3.2.4 Sõnumi formaat ... 27
3.3 SML ... 28
3.3.1 DNS päringu kuju ... 28
3.3.2 CNAME ... 29
3.3.3 NAPTR ... 30
3.3.4 SML serveri implementatsioon ... 30
3.3.5 Arhitektuur ... 31
3.3.6 Testimine ... 31
3.4 SMP ... 32
3.4.1 Teenuse grupp ... 32
3.4.2 Teenuse Metaandmed ... 33
3.4.3 Phoss SMP ... 33
3.4.4 Harmony SMP ... 35
3.4.5 SMP serveri implementatsioon ... 36
3.4.6 Arhitektuur ja arendusmetoodika ... 36
3.4.7 XML C14N ... 37
3.4.8 XMLDSIG Digest ... 38
3.4.9 XMLDSIG SignatureValue ... 39
3.4.10 Testimine ... 39
4 Jõudlustestimine ... 40
4.1 Metoodika ... 40
4.2 Näidislahenduse SMP/SML ... 41
4.3 Phoss SMP ... 42
4.4 Harmony SMP ... 43
4.5 Järeldused ... 43
Kasutatud kirjandus ... 45
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks ... 50
Lisa 2 – Käsklused ... 51
Lisa 3 – PMode sender konfiguratsioon ... 52

Lisa 4 – WS Plugin näidissõnum ..... 56
Lisa 5 – Sõnumi saaja metaandmed ..... 58

# Jooniste loetelu

Joonis 1. Nelja nurga mudel eDelivery ning X-Tee kontekstis ..... 13

Joonis 2. Esimese etapi lõpptulemus - Kahe osapoole vahel vahetatakse sõnumit staatiliselt. ..... 26

Joonis 3. Teise etapi lõpptulemus - Sõnumi saatja alustab dünaamilist otsingut ja saab veateate. ..... 28

Joonis 4. Harmony juurdepääsupunkti poolt tekitatud CoreDNS logid ..... 29

Joonis 5. Algoritm CNAME päringu koostamiseks ..... 29

Joonis 6. Algoritm NAPTR päringu koostamiseks ..... 30

Joonis 7. Kolmanda etapi lõpptulemus - Sõnumi saatja alustab dünaamilist otsingut, saab SML-ist vastuse ning siis veateate. ..... 32

Joonis 8. Harmony juurdepääsupunkti poolt tehtav päring SMP-sse metaandmete saamiseks ..... 33

Joonis 9. Phoss SMP teenuse metaandmete registreerimine ..... 35

Joonis 10. Harmony SMP teenuse metaandmete registreerimine ..... 36

Joonis 11. Funktsionaalselt samaväärsed XML-id aga erinev kuju ..... 38

Joonis 12. Neljanda etapi lõpptulemus - sõnumi saatja saadab sõnumi, kasutades dünaamilist otsingut, sõnumi saajale. Vasakul saaja, paremal saatja ..... 39

# Tabelite loetelu

Tabel 1. Kujutis (mapping) SMP väärtustest juurdepääsupunkti väärtustesse ..... 34

Tabel 2. ab jõudlustestimise mall ..... 41

Tabel 3. Näidislahenduse SMP metaandmete API jõudlus ..... 41

Tabel 4. Näidislahenduse SML DNS-i jõudlus ..... 42

Tabel 5. Phoss SMP jõudlustestide tulemused ..... 42

Tabel 6. Harmony SMP jõudlustestide tulemused ..... 43

Tabel 7. SMP jõudlustestide võrdlus ..... 44

# 1 Sissejuhatus

Euroopa Liit on endale seadnud eesmärgiks aastaks 2050 saavutada kliimaneutraalsus. Selle eesmärgi täitmiseks on Euroopa Liidus loodud erinevaid rahastusvahendeid. Üheks selliseks vahendiks on Euroopa ühendamise rahastu (CEF), mis investeerib kolme peamisesse suunda: energeetika, transport ja digivõrgud. [1]

CEF transport algatas 2023. aastal eFTI4EU projekti, mille eesmärgiks on standardiseeritult digitaliseerida veoinfo vahetus pädevate asutuste ning vedajate vahel lähtudes Euroopa Parlamendi määrusest 2020/1056 [2].

eFTI4EU projekti üks suuremaid tulemusi on eFTI näidislahenduse loomine. Näidislahendus sisaldab minimaalset koodibaasi, mis on vastavuses eFTI4EU määrusega ning mida arendavad projektis osalevad liikmesriigid. Loodud tulemust saavad kasutada liikmesriigid oma riikliku lahenduse alusena. [3]

## 1.1 eFTI taust

eFTI4EU näidislahendus sisaldab kolme põhilist eFTI arhitektuuri komponenti - eFTI värav, eFTI näidisplatvorm ja pädevate asutuste juurdepääsupunkt. [3]

eFTI värav on süsteemi keskne osa, millega suhtlevad nii eFTI platvormid kui ka pädevad asutused. Väravaid haldavad kõik liikmesriigid ise, ning on kohustatud pakkuma seda avaliku teenusena transpordiettevõtetele ja pädevatele asutustele. Värava peamine roll on andmete ning päringute edastamine osapoolte vahel. Väravad peavad olema võimelised omavahel päringuid vastu võtma. Selleks, et standardiseerida suhtlus väravate vahel on kasutusele võetud eDelivery andmevahetuslahendus.

Platvorm täidab eFTIs andmete looja ja haldaja rolli. Platvormi kaudu on võimalik luua eFTI andmehulki, mis kirjeldavad veost ning selle liikumist. Platvormid on üldjuhul eraettevõtted, kes soovivad muuta oma veoandmed kättesaadavaks pädevatele asutustele

läbi eFTI värava. Platvorm saab registreerida veose metaandmeid väravasse, seeläbi muutes need kättesaadavaks kõigile väravatele eFTI võrgustikus. Värav saab pärida ainult alamhulka eFTI andmetest, mis sisaldab vaid neid andmeid, mida pädev asutus seaduslikult peaks kätte saama. Seeläbi minimeeritakse liigsete andmete edastamist. eFTI andmete alamhulgad on defineeritud Euroopa Liidu määrusega 2024/2024 [4]

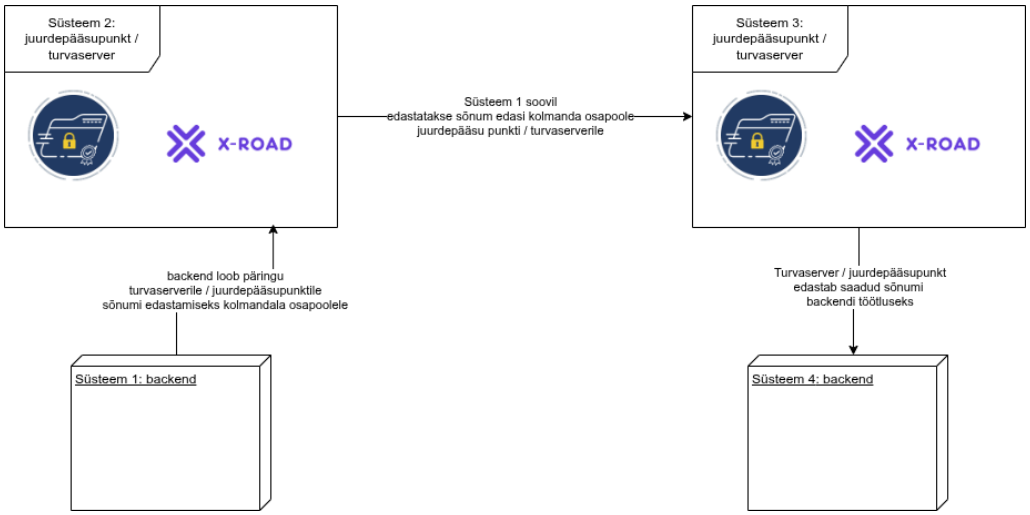
Pädevate asutuste juurdepääsupunktid ehk Authority Access Points (AAP) võimaldavad huvitatud osapooltel pärida eFTI andmeid. Vastavalt liikmesriigi vajadustele võib AAP liidestus väravaga olla mistahes sugune. Eesti kontekstis on võimalik kasutada X-Tee sõnumeid, et pärida andmeid eFTI keskkonnast. [5] Kuna AAP-il pole otsest kokkupuudet eDelivery-ga siis selle lõputöö raamidest jääb AAP välja.

## 1.2 eDelivery taust

eDelivery on avatud spetsifikatsioonidel põhinev infrastruktuur ja protokoll, mis kasutab AS4 (Applicability Statement 4) standardit sõnumite vahetamiseks. See on nagu digitaalne postisüsteem, mis tagab, et sõnumid liiguvad ühest punktist teise kontrollitud ja krüpteeritud viisil. [6]

Sarnane infrastruktuur sõnumivahetuseks on kasutusel ka Eestis X-Tee näol, mis võimaldab turvalist ja tõestusväärtust tagavat internetipõhist andmevahetust riigiasutuste vahel ja erasektoriga. [7]

Arhitektuursel tasandil on eDelivery ja X-Tee sarnased kuna mõlemad põhinevad nelja nurga mudelil (joonis 1). Selle mudeli põhiidee on, et infosüsteemid ei vaheta andmeid omavahel otse. Selle asemel on süsteemid ühendatud juurdepääsupunktide kaudu, mis rakendavad samu tehnilisi spetsifikatsioone ning suudavad seetõttu omavahel suhelda. [8]




Joonis 1. Nelja nurga mudel eDelivery ning X-Tee kontekstis

Sõnumite saatmisel on oluline et saatja teaks saaja metaandmeid. Need andmed on näiteks saaja aadress, sertifikaadid, nimi jms. Saaja metaandmeid on võimalik leida kahel moel: dünaamiliselt ja staatiliselt. Staatilise metaandmete leidmise korral on vajalik info seadistatud saatja andmebaasis ning kättesaadav kohe. Dünaamilise leidmise korral peab andmeid küsima kolmanda osapoole süsteemist. eDelivery-s kutsutakse neid süsteeme teenuse metaandmete avaldaja (SMP) ning teenuse metaandmete leidja (SML). [6]

## 1.3 SMP ja SML taust

SMP on nagu telefoniraamat, mis sisaldab teavet osalejate kohta: nende aadressid, milliseid andmeid nad pakuvad ja sertifikaadid. Iga osaleja eDelivery võrgus haldab oma SMP-d, mis tagab, et süsteem jääb detsentraliseerituks. [6]

SML on ainus tsentraliseeritud eDelivery komponent, mis toimib otsingumootorina. See aitab juurdepääsupunktidel leida õige SMP, kust nad saavad teavet sõnumi saaja kohta. SML põhineb DNS-il, mis tõlgendab organisatsiooni identifikaatori konkreetseks SMP aadressiks. [6]

Näide SMP ja SML kasutamisest eFTI kontekstis:

1. Eesti veoettevõte registreerib veose digitaalselt oma eFTI platvormi.
2. Platvorm peab veose metaandmed edastama Eesti eFTI väravale kuid ei tea värava metaandmeid. Tehakse DNS päring SML-i nimega EE-GATE.

3. SML tagastab eFTI värava SMP serveri aadressi.
4. Platvormi eDelivery juurdepääsupunkt teeb päringu SMP-le, et saada värava metaandmed.
5. Saadud metaandmetest koostatakse päring, mis saadetakse eFTI värava juurdepääsupunkti.
6. Värav salvestab veose andmed.

## 1.4 Probleem

eFTI4EU konsortsiumi poolt loodud näidisarhitektuuris on mõningaid puudujääke. Nende puuduste hulgast on näiteks eDelivery SMP/SML kasutamine platvormi ning värava vahel. Tänane lahendus võimaldab eDelivery põhist suhtlust värava ja platvormi vahel kasutades staatilist konfiguratsiooni, mis ei vasta Euroopa Liidu regulatsioonile 2024/1942 artikkel 9 (3) [9]. Praegune lahendus töötab sarnaselt sellele kuidas väravad omavahel suhtlevad - staatiliselt.

eFTI Regulatsioon hakkab kehtima täies ulatuses 9. juulil 2027. Selleks ajaks peab iga liikmesriik võimaldama pädevatel asutustel ligipääsu eFTI väravatele ning tegema võimalikuks registreerida veoseid eFTI süsteemi. [10] Valmisoleku saavutamiseks on oluline, et platvormid oleksid võimelised saatma veoandmeid kasutades eDelivery SMP/SML teenuseid.

## 1.5 Eesmärk

Käesoleva lõputöö eesmärgiks on lahendada eelnevas peatükis mainitud probleem. Autor on võtnud eesmärgiks uurida, kuidas töötab eDelivery dünaamilise leidmise mehhanism. Oluline on luua PoC lahendus, mis võimaldaks eDelivery dünaamilist otsimist. Tarkvaralise lahenduse puhul on väga tähtis saavutada võimalikult hea jõudlus.

Teiseks eesmärgiks on võtta näidislahenduse tulem kasutusele erinevates eFTI testkeskkondades. See võimaldab huvitatud osapooltel testida eFTI platvormi ja värava vahelist integratsiooni täies ulatuses, nii nagu see on kirjeldatud eFTI regulatsioonis 2024/1942 artikkel 9 (3) [9].

Kolmandaks eesmärgiks on luua juurde dokumentatsiooni ja kasulikku infot eDelivery

ning selle süsteemide kohta. Täna on väga raske leida häid praktilisi näiteid eDelivery kasutamisest. Selle lõputööga soovib autor parandada kasulike materjalide hulka.

## 1.6 Metoodika

TODO, kirjuta siia metoodikast

# 2 eDelivery valmiskomponendid

Hakates looma uut süsteemi, mis kasutab andmete vahetamiseks eDelivery raamistikku, peab arendaja leidma enda projekti jaoks sobilikud valmiskomponendid, mis võimaldavad eDelivery sõnumeid osapoolte vahel saata.

eDelivery standard on tarnija- ja platvormineutraalne, mis tähendab et üks kindel pakkuja ei kontrolli süsteemi arhitektuuri, ning kõigil on võimalik luua oma kohandatud lahendus. See tähendab seda, et täna on turul väga palju erinevaid eDelivery tooteid ja teenuseid, mille vahel saab valikuid teha [11].

Selle lõputöö raames on vaja leida ja kasutusele võtta kolme tüüpi eDelivery lahendusi. Nendeks on:

1. eDelivery juurdepääsupunkt (AP)
2. eDelivery teenuse metaandmete avaldaja (SMP)
3. eDelivery teenuse metaandmete leidja (SML)

Euroopa komisjoni eDelivery tutvustaval veebilehel on välja toodud abistavaid linke ja juhendeid nendele, kes soovivad eDeliveryt kasutusele võtta [11]. Nende linkide seas on viide veebilehele, mis aitab arendajatel leida ja võrrelda AS4 tarkvara või teenuseid, mis on läbinud eDelivery vastavustestimise [12].

Autori eesmärk on luua lahendus, mis töötaks lokaalselt, ning millel ei ole liigseid väliseid sõltuvusi. Selle tõttu on autor otsustanud kasutada ainult eDelivery tooteid, mis on kättesaadavad tasuta, ning mille lähtekood on avalik. See tähendab, et eDelivery teenused ning teenusepakkujaid see uurimustöö ei analüüsi.

## 2.1 eDelivery juurdepääsupunkt (AP)

Kõik täna saadaval olevad eDelivery juurdepääsupunktid on kirjeldatud Euroopa komisjoni vastaval veebilehel [13]. Seal on nimekiri olemasolevatest lahendustest ning nende

tulemustest eDelivery vastavustestimisel. Oluline näitaja on ka testi läbiviimise kuupäev. Kuna eDelivery standard on pidevas muutuses on oluline, et lahendused järgiksid kõige viimast versiooni.

### 2.1.1 Domibus

Nimekirjas esimene lahendus, mis on tasuta, ning mille lähtekood on avalik järgides EUPLv1.2 litsentsi [14]. Domibusi koodivaramu sisaldab kõiki eDelivery osasid - juurdepääsupunkt ja SMP server.

Domibus on Euroopa komisjoni poolt hallatud, AS4 eDelivery juurdepääsupunkti näidislahendus. Tänaseks on sellest välja kujunenud nii-öelda etalonlahendus - seda kasutatakse uute lahenduste põhjana.

Analüüsides Domibusi koodibaasi on tegu üpris suure projektiga. Selleks, et projekti suurust paremini hinnata leidis autor koodibaasi ridade arvu, kasutades käsklust 1. Käskluse tulemus oli 428302 rida koodi.

2024 aastal kirjutati analoogne lõputöö eDelivery ning eFTI teemal, mis käsitleb samamoodi eFTI värava ning platvormi info vahetust kasutades eDelivery süsteemi [15]. Selles töös leidis autor, et Domibusi seadistamine dockeri keskkonnas kaasnes probleemidega - Tomcati veebiserverit ei käivitatud automaatselt. Alternatiivina prooviti kasutada docker hubis valmislahendusena leiduvat Domibusi konteinerit. Lähemalt uurides oli tegu väga vana Domibusi versiooniga, mida polnud kaua uuendatud.

Selle uurimustöö metoodika nõuab, et tulemus oleks võimalikult kasutaja ning arendaja sõbralik (1.6). Loodav lahendus peab olema kindlalt käivitatav dockeri keskkonnas, selleks et tagada vajalik kvaliteet. Kuna Domibus seda ei võimalda kukub see lahenduste kandidaatide hulgast välja.

### 2.1.2 Harmony

Harmony juurdepääsupunkt on avatud lähtekoodiga tarkvara, mille on välja töötanud Põhjamaade Interoperatiivsuse Instituut (NIIS) ning mis põhineb Euroopa Komisjoni Domibus eDelivery lahendusel. See on AS4 protokollil põhinev juurdepääsupunkt, mida kasutatakse turvaliseks ja standarditud andmevahetuseks eDelivery võrgus. [16] NIIS on

muuhulgas ka see organisatsioon, mis töötab eesti X-Tee süsteemi arendamisega.

Võrreldes Domibusi koodibaasiga on Harmony access point kood [17] kõvasti mahukam. Lugedes koodi ridu käsklusega 1, saame 741098, mis on umbes 1.7 korda rohkem kui Domibus.

Kuna Harmony põhineb Domibusi projektil, võtab see endaga kaasa suure osa selle projekti kompleksusest. Kuigi Harmony arendajad on seda proovinud parandada, luues nn *common* koodivaramu [18]. See koodivaramu sisaldab kasulikke skripte Harmony juurdepääsupunkti ja SMP serveri ülesseadmiseks.

Kõige suurem kvaliteet Harmony juurdepääsupunktil, võrreldes Domibusiga on see, et lahendus on arendajate poolt hästi konteineriseeritud, mis muudab selle juurutamise kõvasti lihtsamaks. Konteineriseeritud lahendus on leitav Docker Hubist [19].

Harmony projekt on väga hästi dokumenteeritud ning sisaldab kirjeldusi iga projekti osa kohta. Samuti on olemas dokumentatsioon dünaamilise otsingu seadistamiseks, mis on selles lõputöös väga oluline.

### 2.1.3 Phase4

Phase4 [20] on vabavaraline projekt, mis võimaldab implementeerida erinevaid AS4 põhinevaid kasutusjuhte. Projekt on peamiselt ehitatud ühe arendaja poolt - Philip Helger, kes on Peppoli ning eDelivery maastikul töötanud juba mitukümmend aastat [21].

Phase4 on võimalik kasutada kui teeki, mis suudab võtta vastu ja saata AS4 sõnumeid osapoolte vahel. AS4 üks profiil - eDelivery on ka Phase4 poolt implementeeritud. Selle projekti koodi ridade arvu on raske analüüsida, kuna see sisaldab peale eDelivery veel teisi funktsionaalsuseid. See tähendab reaalsete ridade arv võib olla petlik. Sellegipoolest on ridade arv 495735 käsklusega 1.

Arvestades, et praegune eFTI süsteem ning paljud teised turul kasutatavad lahendused põhinevad Domibusil, on Phase4 kasutamine testimise ja ühilduvuse seisukohast keerulisem. Seetõttu kukub Phase4 kasutamine selle lõputöö raamidest välja.

## 2.2 eDelivery teenuse metaandmete avaldaja (SMP)

Nimekiri täna saadaval olevatest SMP serveritest on Euroopa komisjoni lehel [22]. Iga kirje kohta on saadaval testimise raport, mis kirjeldab testimise kuupäeva ning ühiktestide tulemusi - kas on läbitud või ei ole.

### 2.2.1 Phoss SMP

Phoss SMP on täielik SMP-server, mis toetab nii Peppol SMP 1.x spetsifikatsiooni kui ka OASIS BDXR SMP 1.0 ja 2.0 spetsifikatsioone, mis on kasutusel eDelivery SMP-s [23]. See sisaldab haldusliidest (GUI) ning XML-, SQL- või MongoDB põhist andmebaasi, mis lihtsustab haldusprotsesse. Muuhulgas oli Phoss SMP esimene, mis vastas CEF eDelivery nõuetele [24].

Phoss SMP looja on juba eelnevas peatükis mainitud Philip Helger.

Selle projekti tugevus seisneb selles, et lahendus on väga hästi konteineriseeritud ning lihtsasti kättesaadaval Docker Hubis [25]. Samuti on lahendust võimalik kasutada misiganes eDelivery juurdepääsupunkti jaoks.

Projektis on kokku 315850 rida koodi, mis on saadud käsklusega 1. Arvestades, et see SMP implementeerib peale eDelivery OASIS BDXR-i spetsifikatsiooni veel Peppol SMPd, on suurem ridade arv õigustatud.

### 2.2.2 Harmony SMP

Harmony SMP [26] on välja töötanud, ning seda haldab sama organisatsioon, kes vastutavad Harmony juurdepääsupunkti eest (NIIS). Tarkvara põhineb Euroopa Komisjoni loodud avatud lähtekoodiga projektil DomiSMP [27]. DomiSMP on eDelivery SMP tarkvara näidislahendus, mida on võimalik kasutada teiste lahenduste põhjana.

Harmony SMP koodibaas sisaldab 188220 rida koodi. Võrreldes DomiSMP näidislahendusega [28], mille ridade arv on 216789 siis saame et Harmony SMP on umbes 13% väiksem vastavalt käsule 1

Harmony SMP suur miinus seisneb selles, et seda ei ole konteineriseeritud, mis muudab tarkvara kasutamise raskemaks. [18]

## 2.3 eDelivery teenuse metaandmete leidja (SML)

eDelivery SML-i pakutakse täna teenusena Euroopa Komisjoni poolt kõikidele huvitatud osapooltele. Küllaga on hetkel teenusega ühinemine võimatu, kuna ajutiselt on piiratud uute süsteemide integreerimine SML-i [29].

Komisjoni poolt hallatud eDelivery SML teenus põhineb DomiSML projektil [30]

### 2.3.1 DomiSML

DomiSML on Euroopa Komisjoni poolt pakutav vabavaraline näidislahendus. DomiSML lisab, uuendab ja kustutab DNS-is infot osalejate SMP asukohtade kohta, suunates saatjad õige SMP-ni. See põhineb eDelivery BDXL-profiilil ja PEPPOL SML-i spetsifikatsioonil [30] [31].

DomiSML tarkvara ei ole konteineriseeritud, mis muudab selle väga raskesti kasutatavaks. Samuti sisaldab dokumentatsioon katkiseid linke, tarkvara kasutamise kohta.

DomiSML on siiamaani kõige väiksem tarkvaraprojekt, mis sisaldab 176307 rida koodi vastavalt käsule 1.

Autor proovis käivitada DomiSML tarkvara lokaalselt enda arvutis. Tarkvara käima panekul esines palju erinevaid takistusi. README.md failis leidus viide juhendile, kuidas projekti käivitada. Avades viite veebilehitsejas selgus et link oli katki ning dokumentatsiooni enam ei eksisteerinud.

DomiSML projektis leidus ka *bdmsl-springboot* nimeline alamprojekt, millel oli olemas käivitamisjuhised. Analüüsides juhist lähemalt selgus, et enne käivitamist tuli läbida kahesammuline konfigureerimise protsess. Konfiguratsiooni üks osa oli *MySQL* andmebaasi seadistamine. Kusjuures dokumentatsioon ei täpsustanud millist MySQL versiooni tarkvara ootab.

DomiSML projekti keerukuse ning dokumentatsiooni nappuse tõttu otsustas autor sinna aega mitte investeerida.

# 3 Näidislahenduse loomine

Käesoleva lõputöö eesmärgiks on luua töötav näidislahendus, mis demonstreerib eDelivery dünaamilise otsimise mehhanismi kasutust. Lõppeesmärgini jõudmiseks tegi autor neljaetapilise plaani:

1. Saada tööle staatiline eDelivery konfiguratsioon.
2. Konfigureerida juurdepääsupunkt ümber, selleks et võimaldada dünaamilist otsingut.
3. Luua DNS tarkvara, mis suudab lahendada juurdepääsupunkti poolt tehtavaid päringuid SML-ile.
4. Saada tööle SMP server, mis suudab vastata juurdepääsupunkti päringutele.

## 3.1 Staatiline eDelivery konfiguratsioon

eDelivery staatilise konfiguratsiooni puhul vahetavad juurdepääsupunktid infot otse, ilma kolmanda osapooleta. See on tehniliselt vähem keerukas lahendus, kui dünaamiline otsing, ning selle tõttu ka hea esimene samm lõppeesmärgini jõudmiseks.

### 3.1.1 Juurdepääsupunktide konteinerite seadistamine

Näidislahendus eeldab, et kaks juurdepääsupunkti suudavad omavahel infot jagada. Sellest tulenevalt on vaja käivitada kaks instantsi juurdepääsupunkti tarkvara.

Tarkvara käivitamise lihtsustamiseks, ning üheselt igas masinas samamoodi töötava lahenduse loomiseks kasutab näidislahendus Dockeri konteinereid. Kuna käivitada tuleb mitu erinevat konteinerit, siis kasutab näidislahendus *docker compose* mehhanismi. Compose võimaldab käivitada palju erinevaid konteinereid koos kindla konfiguratsiooniga, kasutades ainult ühte definitsioonifaili: *compose.yml*

Harmony eDelivery juurdepääsupunkt on hästi konteineriseeritud ning dokumenteeritud tarkvara mis on saadaval Docker Hub-is. Seal on olemas ka dokumentatsioon kuidas kasutada Harmony konteinereid compose failis. Sealsete juhiste kohaselt nõuab üks

Harmony instants ka MySQL andmebaasi. [19] See tähendab, et kahe Harmony instantsi jooksutamine nõuab kokku nelja konteinerit.

Näidislahenduse *compose.yml* fail sisaldab kahte juurdepääsupunkti nimedega *sender* ja *receiver*, vastavalt nende tööpõhimõtetele, ning ka kahte MySQL andmebaasi analoogsete nimedega.

### 3.1.2 Juurdepääsupunkti seadistused

Juurdepääsupunkte on võimalik seadistada kahest kohast - keskkonnamuutujad ja veebipõhine kasutajaliides.

Näidislahenduses on *compose.yml* lisatud paar keskkonnamuutujat, mida Docker Hubis olevas näites ei olnud. Need lisandused võimaldavad juurdepääsupunkti õige toimimise. Keskkonnamuutujatest kõige olulisemad on LB_TLS_TERMINATION ja SECURITY_-*_PASSWORD.

**Keskkonnamuutujad**

LB_TLS_TERMINATION muudab juurdepääsupunkti kättesaadavaks pordil 80 (http), mis muudab digitaalsete sertifikaatide haldamise lihtsamaks, kuna ei pea arvestama TLS sertifikaatidega.

SECURITY_*_PASSWORD võimaldab kasutajal, läbi veebiliidese, laadida alla, sirvida ning muuta juurdepääsupunkti privaatseid võtmeid ning avalikke sertifikaate. Ligipääs avalikkele sertifikaatidele on oluline selleks, et neid saaks jagada juurdepääsupunktide vahel, kas staatiliselt või dünaamiliselt.

**Veebipõhine kasutajaliides**

Läbi kasutajaliidese on võimalik konfigureerida juurdepääsupunkti äriloogika põhiseid seadeid. Nende hulgas näiteks digitaalsed sertifikaadid, pluginad ja PMode fail.

PMode fail defineerib juurdepääsupunktide vahelised ühendused ning nende detailid. Tegu on XML dokumendiga, mis kirjeldab näiteks: teise juurdepääsupunkti nime, URLi ja teenuseid.

pluginatest kõige olulisem on selle lõputöö jaoks WS plugin, mille abil on võimalik juurdepääsupunktile saata sõnumeid, mis edastatakse kolmandatele osapooltele.

Loodava lahenduse üks olulisemaid tugevusi on võimaldada väga head arenduskogemust. Selle eesmärgi saavutamiseks on oluline et arendaja saab süsteemi oma masinas tööle panna ilma manuaalse konfiguratsioonita. Näidislahenduses on veebipõhine konfiguratsioon ära automatiseeritud kasutades *bash* skripte, mis sisestavad vajaminevad väärtused automaatselt läbi juurdepääsupunkti REST liidese.

Loodud *bash* skriptid ja vajaminevad konfiguratsioonifailid on liigutatud konteinerite sisse, ning neid käivitatakse konteineri jooksutamisel. Jooksutamise eest vastutab `entrypoint.sh`, mis põhineb originaalsel failil, mida Harmony konteiner kasutab.

### 3.1.3 Digitaalsete sertifikaatide ja võtmete seadistus

eDelivery sõnumite saatmise turvalisus põhineb digitaalsetel sertifikaatidel ja võtmetel, mida kasutatakse sõnumite allkirjastamiseks ja krüpteerimiseks.

Harmony juurdepääsupunkt genereerib vajalikud võtmed ja sertifikaadid automaatselt esimesel käivitusel.

Selleks et saavutada võimalikult hea arenduskogemus tuleb vastavad sertifikaadid ja võtmefailid ise genereerida väljaspool Harmony süsteemi. See tuleneb vajadusest jagada faile kahe juurdepääsupunkti vahel. Kui failid on kohe saadaval sellisel kujul nagu neid vaja on, siis saab neid lihtsasti konteinerisse paigutada.

Harmony juurdepääsupunkt kontrollib sertifikaatide allkirjastamist ning keelab sertifikaatide kasutamise mis ei ole loodud pädevate sertifitseerimiskeskuste (CA) poolt või mida ei ole seadistatud Harmony `truststore`-i.

CA sertifikaatide saamine on tasuline ning jääb selle lõputöö raamidest välja. Selle asemel kasutas autor ise allkirjastatud sertifikaate. Selliseid sertifikaate Harmony automaatselt ei usalda, aga neid on lihtne luua ja kasutada. Alternatiivina oleks võimalik luua ka juursertifikaat, mis allkirjastaks kõiki teisi sertifikaate. Siis tuleks usaldada ainult juursertifikaati. Küllaga ei anna see meetod olulisi eeliseid ise allkirjastatud sertifikaatide ees.

Ise allkirjastatud sertifikaatide ja privaatvõtmete loomiseks kasutas autor openssl teeki käsklusega 2. Projekt sisaldab ka *bash* skripti, mis loob vajalikud sertifikaadid ning paigutab need projektis õigetesse kaustadesse.

### 3.1.4 PMode seadistus

PMode fail sisaldab detaile pakutavate teenuste kui ka teiste juurdepääsupunktide kohta. eFTI keskkonnas on väga oluline et iga liikmesriigi värav toetaks samasuguseid teenuseid. eFTI näidisarhitektuur [3] sisaldab näidis PMode faile, mida on kasutatud selle lõputöö jaoks alusena.

PMode fail peab sisaldama mitmeid eFTI spetsiifilisi elemente, nende hulgas näiteks `eftiGateService`, mis kirjeldab olemasolevat eFTI teenust. Näide saatja PMode failist on toodud koodinäites 5, kus on kirjeldatud kõik vajalikud eFTI spetsiifilised teenused.

PMode faili uuendamine toimub üldjuhul manuaalselt veebikeskkonnast. Selle projekti raames on selle uuendamine automatiseeritud.

### 3.1.5 WS plugin seadistus

WS plugin võimaldab taustsüsteemidel saata sõnumeid juurdepääsupunktile ning vastupidi. Harmony juurdepääsupunkt on vaikimisi seadistatud kasutama WS pluginat kuid võimaldab lihtsasti lisada ka teisi pluginaid.

Võrreldes teiste pluginatega, mida turul täna leidub, autor ühtegi lihtsamini kasutatavat ei leidnud. Kõige silmapaistvam oli Ademico Software nimelise ettevõtte poolt toodetud REST plugin, mis oli kahjuks tasuline ning kinnise lähtekoodiga toode. [32]

Juurdepääsupunkti veebiliideses peab WS Pluginale määrama kasutaja ning parooli, mille abil on hiljem võimalik teha päringuid. Selle projekti raames on nende väljade määramine automatiseeritud.

### 3.1.6 WS plugina kasutamine

Veebiteenuste klassikalisi standardeid (nagu SOAP ja WSDL) kohtab täna uutes projektides üpris vähe, kuna nende kasutuselevõtt on väga ressursikulukas. Seda eelkõige liigse keerukuse, mahuka XML-struktuuri ja rangete standardite tõttu, mis muudavad süsteemide arendamise ja haldamise kohmakaks. [33]

WS plugina WSDL kirjeldus sisaldab 1479 rida XMLi, mis muudab selle väga mahukaks ning keeruliseks.

Kuna WS päringud toimuvad üle HTTP protokolli ning on oma olemuselt POST päringud mille sisu on XML, siis on võimalik neid päringuid teha ka tavapäraste tööriistatega. See tähendab, et pole vaja kasutada väliseid teeke, mis tihtipeale on aeglased ning sisaldavad turvariske. Küllaga tekitab see ühe probleemi - arendaja peab ise sõnumi koostama WSDL kirjelduse järgi, mille muidu oleks valmis teinud väline WS teek.

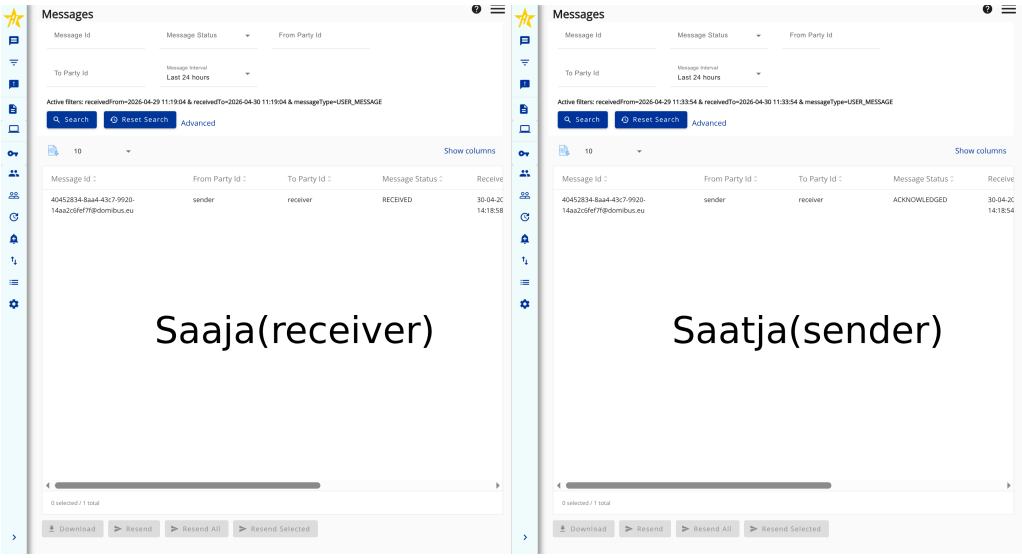
Arvestades probleemi, otsustas autor leida WS plugina näidissõnumi, mida suudaks juurdepääsupunkt töödelda. Näidissõnumit saaks kasutada põhjana teiste sõnumite loomisel.

Näidissõnumi leidmiseks käivitas autor eFTI näidisarhitektuuri projekti ning ka WireShark programmi [34], selleks et jälgida, milliseid andmeid süsteemide vahel saadetakse. eFTI näidisarhitektuuris on värav kasutusel kui taustsüsteem, mis suhtleb Domibus juurdepääsupunktiga kasutades WS plugin ühendust. Kui käivitada päring eFTI identifikaatoritele siis tehakse päring igasse teise ühendatud väravasse. See tähendab et iga värava kohta tehakse üks WS plugin päring, mida on võimalik WireShark programmiga jälgida.

Eelmainitud protsessi tulemusel leidis autor WS pluginale saadetud näidissõnumi, mis on kirjeldatud koodinäites 6.

Kasutades näidissõnumit kirjutas autor ühe failise programmi Kotlinis, mille abil saab saata sõnumeid juurdepääsupunktile.

Selleks, et edukalt lõpetada arendusplaani esimene etapp, saatis autor näidissõnumil põhineva päringu juurdepääsupunktile, mis omakorda edastas selle teisele juurdepääsupunktile. Saatmiseks oli vaja sõnumis muuta saatja ning saaja andmeid, vastavalt eelnevale seadistusele. Joonis 2 on kahe Harmony juurdepääsupunkti admin liides, kus on näha vahetatud sõnumite metaandmed.




Joonis 2. Esimese etapi lõpptulemus - Kahe osapoole vahel vahetatakse sõnumit staatiliselt.

## 3.2 Dünaamiline eDelivery konfiguratsioon

eDelivery dünaamilise konfiguratsiooni puhul toimub sõnumite vahetus juurdepääsupunktide vahel sarnaselt staatilisele ühendusele. Erinevus seisneb selles, et saatev juurdepääsupunkt ei tea ette, kuhu, kellele ja mis moodi tuleb sõnum saata. Selle info leidmiseks tehakse kaks päringut - DNS päring SML-i ja HTTP päring SMPsse.

Selleks, et võimaldada dünaamilist otsingut tuleb juurdepääsupunktid ümber seadistada ning ka sõnumid ära muuta.

### 3.2.1 domibus.properties seadistused

Dünaamilise otsingu võimaldamiseks tuleb lülitada juurdepääsupunkt ümber kasutama SML tsooni ning dünaamilist otsingut. [35, Peatükk 3.2.11 Domibus configuration for OASIS]

Vastavalt Domibusi kasutusmanuaalile saab juurdepääsupunkti äriloogika põhiseid seadeid muuta *domibus.properties* failis. Need samad seaded on saadaval ning muudetavad ka veebipõhises kasutajaliideses. Kuna kõik eelnevad seadistused on tehtud läbi kasutajaliidese automaatselt, otsustas autor samamoodi automatiseerida ka selle seadistuse.

SML tsooniks märkis autor ajutise kohahoidja, mis on dünaamilise otsingu aktiveerimiseks vajalik, kuid ei võimalda edukaid päringuid SML-i.

Selleks et aktiveerida dünaamilist otsingut tuleb sisse lülitada `domibus.dynamicdiscovery.useDynamicDiscovery` seadistus *domibus.properties* failis [18, dynamic_discovery_-configuration_guide.md]. See seadistus on Harmony juurdepääsupunkti puhul hallatav keskkonnamuutujaga `USE_DYNAMIC_DISCOVERY`, mille väärtust saab lihtsasti määrata *compose.yml* failis [19].

### 3.2.2 PMode seadistus

Saatjana ei ole dünaamilises otsinguprotsessis sõnumi saaja ette teada, mistõttu ei ole vaja määrata parameetrit `PMode.Responder`.

Vastuvõtja dünaamilise otsingu konfiguratsioon on sarnane saatja konfiguratsiooniga, lihtsalt rollid on vahetatud: sõnumite saatja pole eelnevalt teada. Mistõttu ei ole vaja parameetrit `PMode.Initiator` määrata [35, Peatükk 3.2.11 PMode configuration for OASIS].

### 3.2.3 Sertifikaadid

Dünaamiline otsing võimaldab leida saaja metaandmeid. Sellest tulenevalt ei ole saatjal enam vaja teada sõnumi saaja avalikku sertifikaati, kuna seda on võimalik SMP serverist küsida. Seega eemaldas autor saatja `truststore`-ist saaja sertifikaadid.

Sõnumi saatja peab usaldama saaja SMP serveri sertifikaati. Selleks seadistas autor SMP serveri sertifikaadi saatja `truststore`-i. See võimaldab saatjal pärida metaandmeid teise osapoole kohta läbi SMP serveri.

Erinevalt saatjast, peab sõnumi saaja usaldama saatja sertifikaate [35, Peatükk 3.2.11 Policy and certificates for OASIS]. Selle tõttu jäi sõnumi saaja `truststore` muutmata.

### 3.2.4 Sõnumi formaat

Dünaamilise otsingu puhul ei ole vaja sõnumi `to` välja konfigureerida. See väli võib sõnumist ka puududa. Oluline on `finalRecipient` välja olemasolu sõnumi `MessageProperties` osas, mis defineerib saaja unikaalse nime [35, Peatükk 3.2.11 Message format for OASIS].

Kõik muud väljad jäävad sõnumis samaks mis olid staatilise otsingu puhul.

Arendusplaani teise etapi lõpuks, saatis autor samasuguse muudetud päringu juurdepääsupunktile mis oli konfigureeritud kasutama dünaamilist otsingut. Selle tulemusel tuvastas saatja juurdepääsupunkt õigesti et tal puudub vajalik info sõnumi edastamiseks ning alustas dünaamilise otsingu protsessi. Selle käigus tehti `CNAME` päring SML serverile, mida teise etapi lõpuks veel ei eksisteerinud. Juurdepääsupunkt andis oodatud veateate päringu ebaõnnestumisest (joonis 3)

```
rtyid-type:unregistered]]
harmony-ap-sender-1    | 2026-04-30 12:31:26,417 [service_account] [default] [d82b7a98-11cb-45a7-a52b-61e21ee992e6@domibus.eu] [] [http-nio-8080-exec-9]  INFO e.d.c.p.p
.d.DynamicDiscoveryPModeProvider:196 - PmodeKey/receiver public certificate not found, starting the dynamic discovery process: [Mandatory field To PartyId is not provid
ed.]
harmony-ap-sender-1    | 2026-04-30 12:31:26,417 [service_account] [default] [d82b7a98-11cb-45a7-a52b-61e21ee992e6@domibus.eu] [] [http-nio-8080-exec-9]  INFO e.d.c.p.p
.d.DynamicDiscoveryPModeProvider:246 - Found [1] dynamic discovery candidates: [[eftiProcess]]. MSHRole: [SENDING]
harmony-ap-sender-1    | 2026-04-30 12:31:26,418 [service_account] [default] [d82b7a98-11cb-45a7-a52b-61e21ee992e6@domibus.eu] [] [http-nio-8080-exec-9]  INFO e.d.c.p.p
.d.DynamicDiscoveryPModeProvider:626 - Perform lookup by finalRecipient type [null] and value [urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4]
harmony-ap-sender-1    | 2026-04-30 12:31:26,418 [service_account] [default] [d82b7a98-11cb-45a7-a52b-61e21ee992e6@domibus.eu] [] [http-nio-8080-exec-9]  INFO e.d.c.p.p
.d.AbstractDynamicDiscoveryServiceImpl:162 - [SMP] Do the lookup by participant identifier [urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4] and scheme [null], docum
ent identifier [eftiGateAction], process identifier [bdx:noprocess] and scheme [tc1]
harmony-ap-sender-1    | 2026-04-30 12:31:50,446 [service_account] [default] [d82b7a98-11cb-45a7-a52b-61e21ee992e6@domibus.eu] [] [http-nio-8080-exec-9] ERROR e.d.c.p.p
.d.AbstractDynamicDiscoveryServiceImpl:200 - Could not fetch metadata from SMP for documentId [eftiGateAction] and processId [bdx:noprocess]
harmony-ap-sender-1    | eu.europa.ec.dynamicdiscovery.exception.DNSLookupException: Lookup [CNAME] for participant [ParticipantIdentifier{identifier='c4', scheme='urn:
oasis:names:tc:ebcore:partyid-type:unregistered'} ] has failed. Lookup result CODE [ 2 ]
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.dns.impl.DefaultDNSLookup.getAllRecordsForType(DefaultDNSLookup.java:233)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.dns.impl.DefaultDNSLookup.dnsRecordNotExists(DefaultDNSLookup.java:166)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.impl.DefaultBDXRLocator.cnameLookup(DefaultBDXRLocator.java:160)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.impl.DefaultBDXRLocator.getUrlForTopDomain(DefaultBDXRLocator.java:131)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.impl.DefaultBDXRLocator.lookupPrivate(DefaultBDXRLocator.java:116)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.locator.impl.DefaultBDXRLocator.lookup(DefaultBDXRLocator.java:141)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryServiceImpl.lookupParticipantSMPUri(DynamicDiscoveryServiceImpl.java:207)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryServiceImpl.getDocumentURI(DynamicDiscoveryServiceImpl.java:193)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryServiceImpl.getFetcherResponseForServiceMetadata(DynamicDiscoveryServiceImpl.java:
180)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryServiceImpl.getServiceMetadata(DynamicDiscoveryServiceImpl.java:113)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.DynamicDiscovery.getServiceMetadata(DynamicDiscovery.java:52)
harmony-ap-sender-1    |        at eu.domibus.core.pmode.provider.dynamicdiscovery.AbstractDynamicDiscoveryServiceImpl.getServiceMetadata(AbstractDynamicDiscoveryServiceImpl.ja
va:216)
```

Joonis 3. Teise etapi lõpptulemus - Sõnumi saatja alustab dünaamilist otsingut ja saab veateate.

## 3.3 SML

Selles arendusetapis lahendatakse ära veateade, mis saadi teise etapi lõpus. Veateade tuleneb sellest, et juurdepääsupunkt ei saa kätte SML serverit, kuhu teha DNS CNAME päring.

Vastavalt eDelivery valmiskomponentide analüüsile, tuleb SML implementeerida koodis otse (vt jaotist 2.3). Selle etapi lõpuks valmib tarkvara, mis käitub kui SML server ning tagastab õiged vastused tehtud päringutele.

### 3.3.1 DNS päringu kuju

SML on oma olemuselt DNS server, kus kasutajad saavad hallata oma SMP serverite DNS kirjeid. Neid kirjeid saavad pärida juurdepääsupunktid nii nagu seda tehti teise etapi lõpus.

Selleks, et uurida milline näeb välja saadetud päring, kasutas autor CoreDNS serverit. CoreDNS on lihtne ja vabavaraline DNS serveri tarkvara, mis on ka konteineriseeritud. Konteineriseerimine võimaldab seda kasutada juba olemasolevas docker compose konfiguratsioonis.

Docker võimaldab konteineritele määrata kohandatud DNS serveri kasutades selleks `dns` nimelist parameetrit. See seadistus eeldab argumendiks IP aadressi, kus asub DNS server. Seda arvestades seadistas autor kõik konteinerid ühte dockeri võrku. Saatja juurdepääsupunkti `dns` väljale määrati CoreDNS serveri IP aadress.

CoreDNS-i on võimalik seadistada kasutades `Corefile`-i, mis kirjeldab DNS päringute töötlemise äriloogikat. Selleks, et uurida milliseid päringuid tehakse juurdepääsupunkti poolt, kasutas autor CoreDNS `log` pluginat, mis logib iga päringu konsooli. Algatanud uue eDelivery päringu kasutades WS pluginat, väljastas CoreDNS logisid (Joonis 4)

```
[INFO] 172.20.0.5:51024 - 48291 "NAPTR IN
    FSPLJSRASNOET76WWBU3VGOATHNBC7HNZ7EUJXU5LKJFNIZVJJSA.testsml. udp
    84 false 4096" NXDOMAIN qr,rd,ra 110 0.0014s
[INFO] 172.20.0.5:49220 - 08273 "CNAME IN B-8
    bb529042b90c77892efd34f7395f504.testsml. udp 42 false 4096"
    NXDOMAIN qr,rd,ra 98 0.0009s
```

Joonis 4. Harmony juurdepääsupunkti poolt tekitatud CoreDNS logid

Logidest saab järeldada, et Harmony juurdepääsupunkt teeb kahte sorti DNS päringuid - NAPTR ja CNAME.

### 3.3.2 CNAME

CNAME kirje puhul on tegemist nimega, mis viitab teisele nimele. Tehniliselt võib seda nimetada aliaseks - üks ja sama aadress on kättesaadav mitme nime kaudu [36].

Selle loogika põhjal peaks Harmony juurdepääsupunkti poolt tehtud B-8bb529042b90c77892efd34f7395f504.testsml CNAME päring vastama SMP serveri URL-ile. Selleks et leida SMP serveri URL, peab teadma kuidas koostada `B-*` räsi.

Algoritm CNAME päringu koostamiseks on pseudokoodis kirjeldatud joonisel 5 [37, Peatükk 2.1.2].

```
"B-"+hexstring(md5(lowercase(ID-VALUE)))+"."+ID-SCHEME+"."+SMLZONE-NAME
```

Joonis 5. Algoritm CNAME päringu koostamiseks

ID väärtus, millest räsi luuakse, vastab `finalRecipient` väljale saadetavas sõnumis, mis eFTI kontekstis on urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4

### 3.3.3 NAPTR

NAPTR-päring on DNS-i kirje tüüp, mis võimaldab reeglipõhiselt teisendada ühe sümbolistringi (näiteks telefoninumbri või domeeninime) konkreetseks teenuseks või aadressiks. Teisendusi kirjeldab vastuses olev regulaaravaldis. Seda kasutatakse peamiselt VoIP-s, et leida õige protokoll ja server, mille kaudu kõnet alustada [38, Sektsioon 1].

Juurdepääsupunkti poolt tehtud NAPTR päringu loomiseks kasutatakse algoritmi, mis on pseudokoodis kirjeldatud joonisel 6 [37, Peatükk 2.1.2].

```
strip-trailing(base32(sha256(lowercase(ID-VALUE))),"=")+"."+IDSCHEME
    +"."+SML-ZONE-NAME
```

Joonis 6. Algoritm NAPTR päringu koostamiseks

ID väärtus on sama mis CNAME päringu puhul.

### 3.3.4 SML serveri implementatsioon

Täieliku SML serveri loomiseks tuleb implementeerida nii NAPTR kui ka CNAME päringute tugi. See võimaldab vastata igale päringule mida juurdepääsupunkt teha võib. NAPTR päringu implementeerimine on eriti oluline, kuna sellega on võimalik kasutada HTTPS-i, erinevalt CNAME päringutest, mis toetavad ainult HTTP protokolli [39].

SML server luuakse samas keeles, milles loodi WS plugina klient rakendus - Kotlin.

**Klite**

Serveri raamistikuks valis autor Klite-i

Klite on modulaarne ja välissõltuvusteta Kotlini veebiraamistik, mille arhitektuuriline fookus on suunatud tarkvara jätkusuutlikkusele ning ökoloogilise jalajälje minimeerimisele. Raamistik tugineb Java sisseehitatud lahendustele ja Kotlini korutiinidele, mis muudab asünkroonsete lahenduste loomise väga lihtsaks. Vaatamata oma kergusele pakub Klite arendajale täisfunktsionaalset ökosüsteemi, kus integreeritud baasmoodulid ja läbipaistev

koodibaas lihtsustavad nii rakenduse loomist kui ka selle pikaajalist hooldamist [40].

### 3.3.5 Arhitektuur

Arhitektuuriliselt koosneb SML server kahest osast - DNS ja Register.

**DNS** on alguspunkt, kuhu kõik SML päringud alguses jõuavad. DNS moodul kasutab `dnsjava` teeki päringute ja vastuste haldamiseks.

Dnsjava on Java programmeerimiskeeles loodud terviklik DNS-protokolli teostus, mis toetab peaaegu kõiki kirjetüüpe, päringute tegemist, tsoonisiirdeid, dünaamilisi uuendusi ja DNSSEC-i valideerimist [41].

DNS server kuulab päringuid ning vastab neile sõltuvalt sellest mis tüüpi need on. SML puhul on vaja vastata ainult kahte sorti päringutele. Päringule leiab õige vastuse register.

**Registri** ülesanne on hoida DNS päringute vastuseid. SML kontekstis peab register hoidma võti-väärtus kirjeid CNAME ja NAPTR päringute jaoks. Mõlema puhul on võtmeks ärireeglite kohaselt räsitud `finalRecipient` (vt 3.3.2, 3.3.3). CNAME puhul on väärtuseks SMP kanooniline nimi. NAPTR puhul on väärtuseks regulaaravaldis, mis suunab SMP serverisse.

### 3.3.6 Testimine

Tulemuse testimiseks kirjutas autor ühikteste DNS mooduli ja räsimise äriloogika jaoks.

SML serveri funktsionaalseks testimiseks tegi autor sama WS plugin päringu nagu teise etapi lõpus. Seekord oli veasõnum teine - Harmony juurdepääsupunkt ei leia SMP-st metaandmeid. Küllaga on näha et NAPTR päring tehakse edukalt, ning juurdepääsupunkt üritab teha päringut SMP serverile (joonis 7).

```
.d.DynamicDiscoveryPModeProvider:626 - Perform lookup by finalRecipient type [null] and value [urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4]
harmony-ap-sender-1    | 2026-05-03 14:03:46,883 [service_account] [default] [ae26e66e-9e8a-4a60-9a33-aabbafca38c0@domibus.eu] [] [http-nio-8080-exec-7]  INFO e.d.c.p.
.d.AbstractDynamicDiscoveryService:162 - [SMP] Do the lookup by participant identifier [urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4] and scheme [null], docu
ent identifier [eftiGateAction], process identifier [bdx:noprocess] and scheme [tc1]
harmony-ap-sender-1    | 2026-05-03 14:03:46,884 [service_account] [default] [ae26e66e-9e8a-4a60-9a33-aabbafca38c0@domibus.eu] [] [http-nio-8080-exec-7]  INFO e.d.c.p.
omibusProxyServiceImpl:69 - Initialize Domibus proxy.
harmony-ap-sender-1    | 2026-05-03 14:03:46,884 [service_account] [default] [ae26e66e-9e8a-4a60-9a33-aabbafca38c0@domibus.eu] [] [http-nio-8080-exec-7]  INFO e.d.c.p.
omibusProxyServiceImpl:72 - Proxy not required. The property domibus.proxy.enabled is not configured
smp                    | 94115 [Thread-1] INFO SMLServer - Handling query for NAPTR FSPLJSRASNOET76WWBU3VGOATHNBC7HNZ7EUJXU5LKJFNIZVJJSA.testsml.
smp                    | 94335 [4bde-1] INFO RequestLogger - 172.20.0.5 GET /urn:oasis:names:tc:ebcore:partyid-type:unregistered:C4/services/::eftiGateAction: 404 in 8
 ms - Apache-HttpClient/5.2.3 (Java/11.0.28)
harmony-ap-sender-1    | 2026-05-03 14:03:47,563 [service_account] [default] [ae26e66e-9e8a-4a60-9a33-aabbafca38c0@domibus.eu] [] [http-nio-8080-exec-7] ERROR e.d.c.p.
.d.AbstractDynamicDiscoveryService:200 - Could not fetch metadata from SMP for documentId [eftiGateAction] and processId [bdx:noprocess]
harmony-ap-sender-1    | eu.europa.ec.dynamicdiscovery.exception.DNSLookupException: SMP lookup address http://smp/urn%3Aoasis%3Anames%3Atc%3Aebcore%3Apartyid-type%3Au
registered%3AC4/services/%3A%3AeftiGateAction not found - response 404
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.fetcher.impl.DefaultURLFetcher.connect(DefaultURLFetcher.java:222)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.core.fetcher.impl.DefaultURLFetcher.fetch(DefaultURLFetcher.java:168)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryService.getFetcherResponseForServiceMetadata(DynamicDiscoveryService.java:
182)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.service.impl.DynamicDiscoveryService.getServiceMetadata(DynamicDiscoveryService.java:113)
harmony-ap-sender-1    |        at eu.europa.ec.dynamicdiscovery.DynamicDiscovery.getServiceMetadata(DynamicDiscovery.java:52)
harmony-ap-sender-1    |        at eu.domibus.core.pmode.provider.dynamicdiscovery.AbstractDynamicDiscoveryService.getServiceMetadata(AbstractDynamicDiscoveryService.j
va:216)
harmony-ap-sender-1    |        at eu.domibus.core.pmode.provider.dynamicdiscovery.AbstractDynamicDiscoveryService.lookupInformation(AbstractDynamicDiscoveryService.ja
a:170)
harmony-ap-sender-1    |        at eu.domibus.core.pmode.provider.dynamicdiscovery.AbstractDynamicDiscoveryService$$FastClassBySpringCGLIB$$776ed08b.invoke(<generated:
harmony-ap-sender-1    |        at org.springframework.cglib.proxy.MethodProxy.invoke(MethodProxy.java:218)
harmony-ap-sender-1    |        at org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.invokeJoinpoint(CglibAopProxy.java:792)
```

Joonis 7. Kolmanda etapi lõpptulemus - Sõnumi saatja alustab dünaamilist otsingut, saab SML-ist vastuse ning siis veateate.

## 3.4 SMP

Selles etapis, sarnaselt eelmisele, lahendatakse ära saadud veateade. Veateate põhjuseks on see, et Harmony juurdepääsupunkt ei saa vastust SMP serverist, kuna seda ei eksisteeri.

Viimases etapis kasutab autor SMP tarkvara valmiskomponente, et uurida kuidas need töötavad, ning loob saadud teadmiste abil töötava SMP serveri.

SMP serverid peavad avalikustama kahte sorti REST API lõpp-punkte - Teenuse grupp ja Teenuse Metaandmed [42, ptk 5.4].

### 3.4.1 Teenuse grupp

Teenuse grupp kirjeldab mingit kindlat osalejat SMP-s. Osaleja unikaalne tunnus moodustub tema id-st ning id skeemist. eDelivery osaleja tunnus on defineeritud OASIS ebCore Party Id Type spetsifikatsiooni järgi [43], mille kohasel on osaleja id skeemi alguseks alati `urn:oasis:names:tc:ebcore:partyid-type` [44, ptk 2.1], peale mida järgneb skeemikatalogi tunnus.

Üldjuhul peab id väärtus olema registreeritud mõnes skeemikataloogis nagu näiteks iso6523, iso9735 and iso20022. Skeemikataloog `unregistered` määrab funktsionaalselt seda, et osaleja id võib olla ükskõik mis väärtus, ning ei pea kuuluma mõnda üldtuntud skeemikataloogi [44, ptk 2.5]. Näidisprojekti raames on sõnumi saatja kui ka saaja osaleja id skeemiks `urn:oasis:names:tc:ebcore:partyid-type:unregistered`.

Osalejate id väärtusteks märkis autor vastavalt `sender` ja `receiver`. Selle muudatusega koos muutis autor ka SML serveri registri sissekandeid ning WS plugin sõnumi saatja ning saaja väljasid.

Osaleja id väärtuse ning id skeemi vahel võib olla kas üks koolon või kaks koolonit. See tuleneb kahest erinevast standardist, mis mõlemad on eDeliverys lubatud [45, ptk 3.3] [42, ptk 5.4] [44, ptk 2.7]. Küllaga on kahe kooloni kasutamine eelistatud, kuna ilma selleta ei saa kindlalt eraldada id skeemi ning väärtust. Kui iga sõne vahel oleks ainult üks koolon siis ei oleks üheselt mõistetav milline osa tunnusest on skeem ning milline on väärtus [46, Kommentaar 3]. Seda arvestades lisas autor WS plugin sõnumis `originalSender` ja `finalRecipient` väljade ette ühe kooloni.

Peale kõiki muudatusi tehakse Harmony juurdepääsupunkti poolt päringuid SMP serverisse joonisel 8 kirjeldatud kujul.

```
GET /urn:oasis:names:tc:ebcore:partyid-type:unregistered::receiver/
    services/::eftiGateAction
```

Joonis 8. Harmony juurdepääsupunkti poolt tehtav päring SMP-sse metaandmete saamiseks

### 3.4.2 Teenuse Metaandmed

Teenuse metaandmed sisaldavad infot juurdepääsupunkti kui ka tema teenuse kohta. Kindlat teenust iseloomustab tema unikaalne tunnus mis koosneb id-st ning skeemi id-st.

Vastavalt eFTI näidisarhitektuurile [3] on teenuse tunnuseks `::eftiGateAction`. Sellise tunnusega teenust otsitakse SMP-st, kuna WS Plugin poolt saadetud sõnum sisaldab `<Action>` väärtuses `eftiGateAction`. Seda väärtust autor muutma ei hakanud selleks, et säilitada ühilduvus eFTI näidisarhitektuuriga.

### 3.4.3 Phoss SMP

Vastavalt eDelivery SMP tarkvara valmiskomponentide analüüsile (vt jaotist 2.2) on Phoss SMP väga lihtsasti kasutatav SMP serveri tarkvara. Selle tõttu otsustas autor seda ka lähemalt uurida.

Phoss SMP toetab kolme sorti tagasüsteemi - MongoDB, XML ja SQL, mis on kõik kon-

teineriseeritud [24]. Näidislahenduse arendamise jaoks valis autor XML põhise lahenduse, kuna see ei nõua, peale SMP enda, veel ühte lisakonteinerit andmebaasi jaoks. XML taustsüsteemis on kõik salvestatud XML failidesse [25].

Autor käivitas Phoss SMP serveri ning logis sisse veebipõhisesse adminliidesesse. Sealt avanes vaade erinevatele SMP süsteemi osadele, mida on administraatoril võimalik muuta. Konfiguratsioonidest kõige olulisemateks osutusid "Service Groups"ja "Endpoints", mille abil on võimalik registreerida vastavalt osalejaid ja teenuseid.

Registreerinud `receiver` osaleja SMP-sse loodi `/urn:oasis:names:tc:ebcore:partyid-type:unregistered::receiver` nimeline REST API lõpp-punkt, mis väljastab metaandmeid selle osaleja kohta XML kujul.

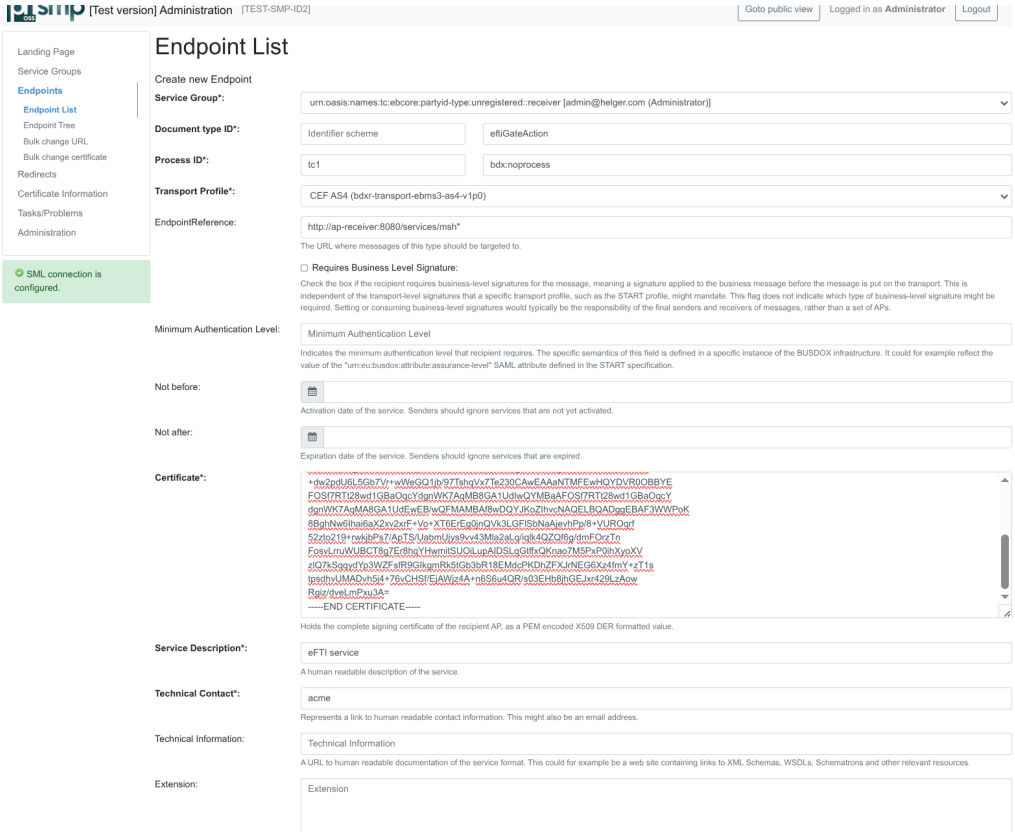
Peale osaleja registreerimist avas autor *Endpoints* menüü, et registreerida eFTI teenust. Teenuse registreerimisel oli vaja sisestada oluliselt rohkem väärtuseid kui osaleja puhul.

Kaks kõige esimest välja teenuse registreerimisel olid **dokumendi id tüüp** ning **protsessi id**. Välja nimed tekitasid segadust, kuna sellenimelisi välju ei ole defineeritud WS plugin sõnumis. Uurimuse käigus selgus, et SMP ning juurdepääsupunkti sõnavara erineb üksteisest, kuigi põhimõtted on samad. Vastavalt AS4 spetsifikatsioonile, saab luua kujutuse SMP väärtustest juurdepääsupunktide väärtustesse [47, ptk 5] (Tabel 1)

Tabel 1. Kujutis (mapping) SMP väärtustest juurdepääsupunkti väärtustesse

| SMP väärtus | Juurdepääsupunkti väärtus |
|---|---:|
| Process | Service |
| Document | Action |

Arvestades leitud erisusi, täitis autor ära kõik kohustuslikud väljad teenuse registreerimiseks (Joonis 9). Küllaga ei olnud võimalik teenust registreerida, kuna süsteem andis veateate - dokumendi tüübi id skeem peab olema defineeritud. Dokumendi tüübi id skeemi eFTI kontekstis ei eksisteeri (vt 3.4.2). Näidislahenduse loomiseks ei ole seega võimalik kasutada Phoss SMP serverit.




Joonis 9. Phoss SMP teenuse metaandmete registreerimine

### 3.4.4 Harmony SMP

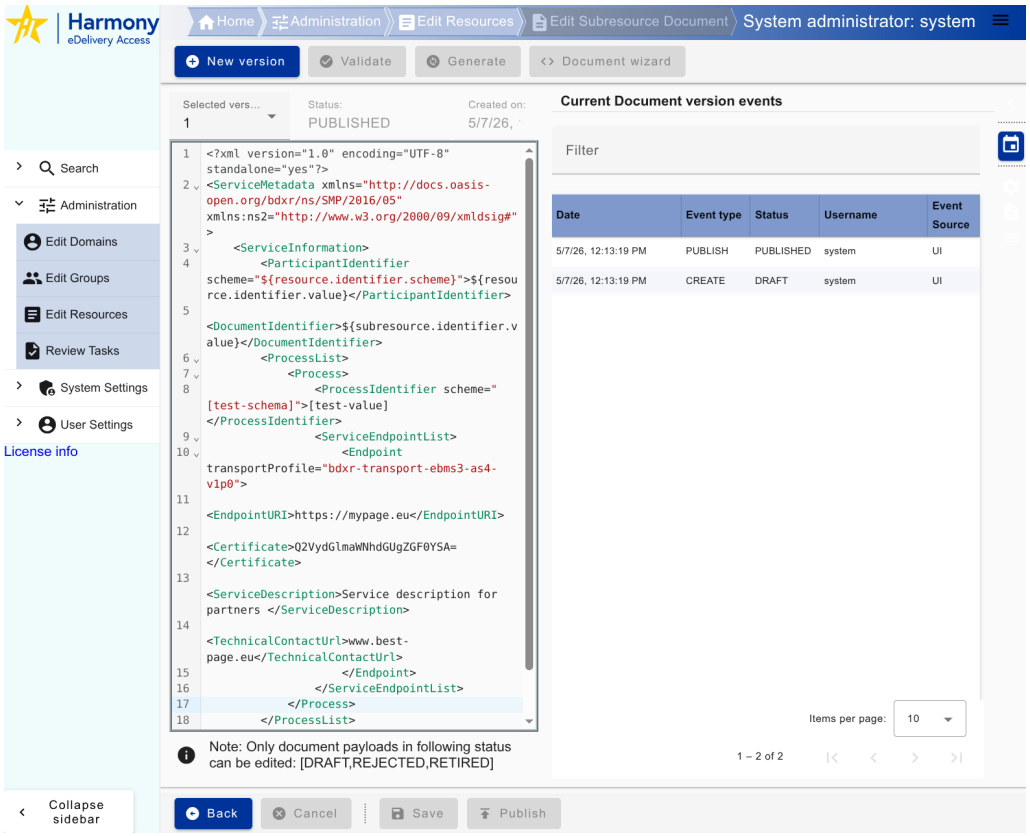
Vastavalt valmiskomponentide analüüsile on Harmony SMP serveri üks suurimaid miinuseid see, et seda pole konteineriseeritud (vt jaotist 2.2). Katsetamise eesmärgil Otsustas autor Harmony SMP ise konteineriseerida.

Autor uuris Harmony SMP koodibaasi ning leidis, et projekt oli juba konteineriseeritud. Küllaga ei olnud võimalik ehitada dockeri pilti sealsete skriptide abil. Põhjus seisnes selles, et Harmony SMP muudatused DomiSMP vastu olid ära lõhkunud SMP serveri konteinerite piltide ehitamise funktsionaalsuse. Autor parandas leitud vead ning sai pildid uuesti ehitama. Loodud pildid laadis autor ka Docker hub keskkonda. [48] [49].

Kasutades dokeriseeritud Harmony SMP serverit, logis autor sisse veebipõhisesse admin liidesesse. Võrreldes Phoss SMP-ga oli Harmony kasutajaliides raskemini arusaadav ning navigeeritav. Teenuse metaandmete registreerimine oli võimalik ainult manuaalselt XML-i muutes (Joonis 10).

Manuaalne XML-i muutmine võimaldas autoril sisestada dokumendi id skeemi vastavalt

eFTI nõetele, ilma et oleks tekkinud veateade. Selle tulemusena oli võimalik koguda näidis XML-e, mille abil on võimalik luua minimaalne SMP implementatsioon näidisprojekti raames.




Joonis 10. Harmony SMP teenuse metaandmete registreerimine

### 3.4.5 SMP serveri implementatsioon

Selleks, et lahendada eelmises peatükis olev veateade metaandmete päringu kohta, tuleb luua REST API lõpp-punkt mis tagastab juurdepääsupunktile teenuse metaandmeid. Päringu kuju on kirjeldatud joonisel 8.

Selleks et hoida projekt lihtsana otsustas autor integreerida SMP REST teenuse SML-iga kokku. See tähendab et Klite haldab nii SMP kui ka SML osa.

### 3.4.6 Arhitektuur ja arendusmetoodika

SMP alamsüsteem koosneb kolmest osast - Metaandmete register, Metaandmete genereerija ja SMP ruuter

**Metaandmete registri** ülesanne on hoida teenuse metaandmeid ning osalejate paare. Ühele

osalejale vastab üks sissekanne metaandmeid XML kujul. Kuna teenus on alati sama (`eftiGateAction`), siis pole vaja hoida võtmetena osaleja-teenuse kombinatsioonina.

**Metaandmete genereerija** ülesanne on genereerida metaandmete XML. Ainukesed muutuvad osad metaandmetes on sõnumi saaja, sõnumi saaja juurdepääsupunkti URL ja saaja digitaalne sertifikaat. Nende abil on võimalik luua teenuse metaandmed, mida võtab vastu juurdepääsupunkt.

**SMP ruuter** võtab vastu HTTP päringuid teenuse metaandmetele ning tagastab küsitud andmed XML kujul, juhul kui need on olemas.

Teenuse metaandmete XML põhjana kasutas autor Harmony SMP serveri poolt loodud andmeid. Võrreldes Phoss SMP-ga ei kasutanud Harmony SMP XML väljade jaoks NS prefiksit. NS prefiksi mitte kasutamine, kui see võimalik on, muudab XMLi lihtsamini loetavaks, mille tõttu autor otsustas selle kasuks.

Kõige keerulisem osa selle projekti juures oli õige metaandmete XMLi loomine, mida suudaks juurdepääsupunkt vastu võtta. Keerukust tekitas XMLDSIG.

XMLDSIG ehk XML-allkirjad pakuvad terviklust, sõnumi autentimist ja allkirjastaja autentimist mis tahes tüüpi andmetele, olenemata sellest, kas need asuvad allkirja sisaldavas XML-failis või mujal [50].

Üldjuhul kasutatakse spetsiifilisi teeke selleks, et allkirjastada XML dokumente. Nende teekide probleem on see, et need on üldjuhul väga aeglased. Nagu näiteks Java XMLSignature mis kasutab XML-i töötlemiseks DOM-i. DOM on väga aeglane algoritm XML-i töötlemiseks [51]. Sellest tulenevalt otsustas autor XML-i allkirjastamiseks klassikalisi XML-i töötlemise tööriistu mitte kasutada.

### 3.4.7 XML C14N

On võimalik, et XML-dokumendid, mis on paljude rakenduste jaoks samaväärsed, erinevad oma füüsilise esituse poolest. Näiteks võivad need erineda oma olemite struktuuri, atribuutide järjestuse ja märgikodeeringu poolest (Joonis 11) [52].

```
<user id="123" status="active">
  <name>Jane Doe</name>
</user>

<user status="active" id="123">
  <name>Jane Doe</name>
</user>
```

Joonis 11. Funktsionaalselt samaväärsed XML-id aga erinev kuju

XML-i Kanooniline kuju on funktsionaalselt samasugustel XML-idel ka samasugune. See võimaldab räsida XML-e samamoodi, olenemata millisel kujul nad algselt olid. Räsimisel on oluline et sisendandmed oleks alati samad, kuna isegi väike muudatus sisendis toob kaasa teise räsi.

Teenuse metaandmete XML-is on kasutatud Canonical XML v1 formaati kanoonilise kuju leidmiseks. See tuleneb `CanonicalizationMethod` välja `http://www.w3.org/TR/2001/REC-xml-c14n-20010315` väärtusest [53].

Lähtudes ülaltoodust, paigutas autor metaandmete XMLi väljad kanoonilisele kujule. Kuna andmed juba on kanoonilisel kujul siis ei ole vaja kasutada mahukaid XMLi teeke, mis muudab koodi lihtsamaks ning kiiremaks.

### 3.4.8 XMLDSIG Digest

Selleks et veenduda, et teenuse metaandmeid pole kuidagi muudetud, luuakse `SHA256` räsi nendest andmetest. Räsimise algoritm on kirjeldatud XML-is viitega `http://www.w3.org/2001/04/xmlenc#sha256`

Räsimiseks tuli paigutada XML-i metaandmete osa `SignedServiceMetadata` osa vahele. Väga oluline oli eemaldada kõik teised XMLi osad räsimisel, sest muidu loodi vale räsi. See tuleneb viitest `http://www.w3.org/2000/09/xmldsig#enveloped-signature` mille kohaselt tuleb eemaldada `Signature` osa täielikult räsi loomisel. [50, ptk 6.6.4]

Loodud räsi paigutati `DigestValue` väljana `SignedInfo` sisse.

### 3.4.9 XMLDSIG SignatureValue

Loodud `SignedInfo` välja peab digitaalselt allkirjastama. Selleks tuleb kasutada `SHA256withRSA` algoritmi, mis tuleb `SignatureMethod` väljast `http://www.w3.org/2001/04/xmldsig-more#rsa-sha256` [54, ptk 3.1.2].

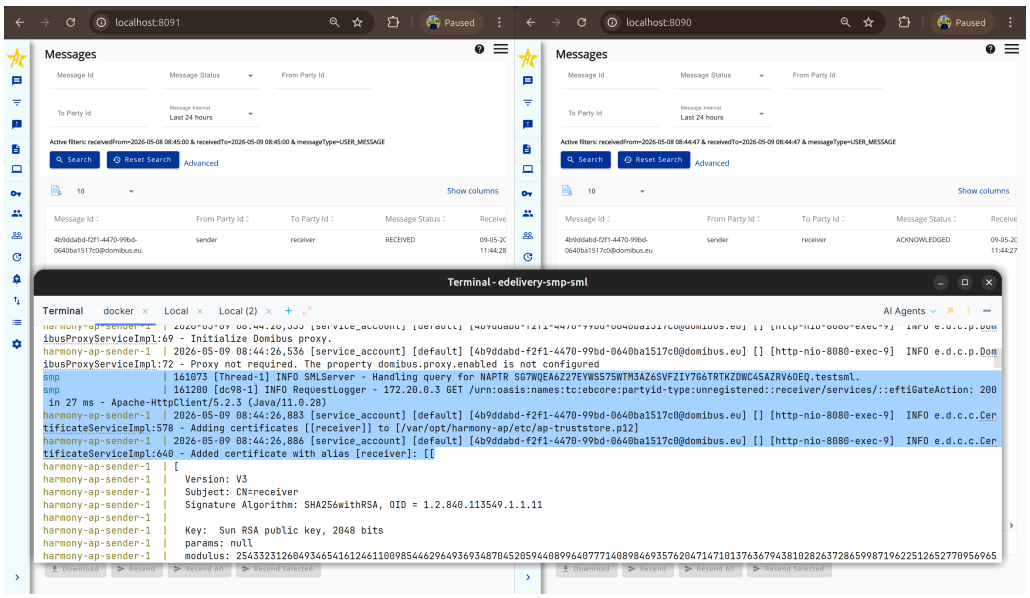
Saadud allkiri tuleb paigutada `SignatureValue` välja sisse [54, ptk 3.1.2].

### 3.4.10 Testimine

Autor kirjutas SMP serveri implementatsioonile ühikteste ning viis läbi funktsionaalse testimise.

Funktsionaalse testimise käigus tegi autor HTTP GET päringu teenuse metaandmetele. Metaandmete vastus on kirjeldatud lisas 7.

Viimaseks saatis autor WS plugin sõnumi saatja juurdepääsupunktile, mis edukalt leidis sõnumi saaja andmed SMP serverist (joonis 12)




Joonis 12. Neljanda etapi lõpptulemus - sõnumi saatja saadab sõnumi, kasutades dünaamilist otsingut, sõnumi saajale. Vasakul saaja, paremal saatja

# 4 Jõudlustestimine

Jõudlustestimise eesmärgiks on saada teada kui kiire ja vastupidav on mingi tarkvalahendus. Selle lõputöö käigus viiakse läbi jõudlustestimine loodud näidislahenduse peal. Testimise skoopi kuuluvad ka Harmony SMP ning Phoss SMP server.

Jõudlustestimise skoopi ei kuulu WS plugin sõnumite saatmine läbi juurdepääsupunkti. Testimise eesmärgiks on teha kindlaks loodud näidislahenduse tulemused võrreldes teiste komponentidega. Sellest tulenevalt ei ole vajalik testida Harmony juurdepääsupunkti jõudlust.

Testimine jaguneb järgnevateks etappideks:

1. Näidislahenduse SMP ja SML testimine
2. Phoss SMP testimine
3. Harmony SMP testimine

## 4.1 Metoodika

Jõudluse testimisel tuleb paika seada millised on testimise sisendid ning millised on võrreldavad väljundid.

Väljundina võrreldakse töödeldud päringute kogust sekundis ning päringute 99% latentsust. Seda tüüpi tulemuste kogumiseks on kõige paremaks tööriistaks Apache Benchmark. Apache Benchmark (ab) on lihtne, kuid võimas käsureatööriist, mida kasutatakse veebiserverite (mitte ainult Apache) koormustestimiseks ja jõudluse mõõtmiseks. See annab kiire ülevaate sellest, kui hästi suudab server tulla toime suure hulga päringutega. [55]

Apache Benchmark moodi lahendus DNS-i maastikul on `dnsperf`. `dnsperf` on tasuta tööriist, mille on välja töötanud Nominum/Akamai (2006–2018) ja DNS-OARC (alates 2019. aastast) ning mis teeb latentsuse ja läbilaskevõime meetrikate kogumise lihtsaks [56].

Mõlema tööriista puhul peab määrama mitu päringut tehakse kokku ning kui palju päringuid tehakse samaaegselt. Autor tegi jõudlustestimiseks argumentide ja tulemuste malli (tabel 2). Päringute arv ja samaaegsus on järjest kasvav. Nii on võimalik võrrelda tarkvara jõudlust erinevatel koormusastmetel.

Tabel 2. ab jõudlustestimise mall

| Päringute arv (-n) | Samaaegsed päringud (-c) | Päringuid sekundis (mediaan) | 99% latentsus |
|---|---|---|---|
| 100 | 10 | | |
| 1000 | 100 | | |
| 5000 | 250 | | |
| 10000 | 1000 | | |

Kõik testimised toimuvad kasutades Intel(R) Core(TM) Ultra 7 255H (16) @ 5.10 GHz protsessorit.

## 4.2 Näidislahenduse SMP/SML

Autor käivitas näidislahenduse SMP/SML serveri ning kogus selle kohta jõudlusandmeid kasutades selleks `ab` ja `dnsperf` tööriistu.

Näidislahenduse SMP serveri jõudlus hinnati kasutades käsklust 3. Tulemused on kirjeldatud tabelis 3.

Tabel 3. Näidislahenduse SMP metaandmete API jõudlus

| Päringute arv (-n) | Samaaegsed päringud (-c) | Päringuid sekundis (mediaan) | 99% latentsus (ms) |
|---|---|---|---|
| 100 | 10 | 3134 | 3.2 |
| 1000 | 100 | 920 | 108 |
| 5000 | 250 | 3220 | 77 |
| 10000 | 1000 | 4920 | 200 |

Näidislahenduse SML serveri jõudlus hinnati kasutades käsklust 4. Tulemused on kirjeldatud tabelis 4.

Tabel 4. Näidislahenduse SML DNS-i jõudlus

| Päringute arv (-n) | Samaaegsed päringud (-c) | Päringuid sekundis (mediaan) | 99% latentsus (ms) |
|---|---|---|---|
| 100 | 10 | 11314 | 1.6 |
| 1000 | 100 | 20221 | 4 |
| 5000 | 250 | 38383 | 2 |
| 10000 | 1000 | 5192 | 19 |

Tulemustest saab järeldada et näidislahenduse SMP moodul kui ka SML moodul suudavad toime tulla väga suure koormusega. SMP serveri jõudlustabel sisaldab anomaaliat tuhande päringu ja saja samaaegse päringu puhul. Autor ei suutnud välja selgitada millest tulenes madalam jõudlus just sellel koormusastmel.

## 4.3 Phoss SMP

Kuna Phoss SMP võimaldab vaid SMP funktsionaalsust siis koguti andmeid ainult selle osa kohta. Autor muutis käsklust 3 ning määras metaandmete URL-iks `http://localhost:8080/urn:oasis:names:tc:ebcore:partyid-type:unregistered::receiver/services/bus::eftiGateAction`. Muudatus tuleneb sellest, et Phoss SMP ei võimalda registreerida teenust millel ei ole skeemi id-d. Jõudlustestimise võimaldamiseks määras autor skeemi id-ks `bus`, kõik teised väärtused jäid näidislahendusega samaks.

Jõudlustestide tulemusi kirjeldab tabel 5.

Tabel 5. Phoss SMP jõudlustestide tulemused

| Päringute arv (-n) | Samaaegsed päringud (-c) | Päringuid sekundis (mediaan) | 99% latentsus (ms) |
|---|---|---|---|
| 100 | 10 | 1245 | 8 |
| 1000 | 100 | 3080 | 32 |
| 5000 | 250 | 1297 | 192 |
| 10000 | 1000 | 1290 | 774 |

Tulemustest saab järeldada, et Phoss SMP suudab taluda suurt koormust hästi. Samuti on näha et serveri jõudlus langeb kiirelt koormuse suurenemisel.

## 4.4 Harmony SMP

Kuna Harmony SMP ei serveeri teenuse metaandmeid URL-i juurelt, tuleb kohandada käsklust 3, muutes SMP URL-iks `http://localhost:8080/smp/urn:oasis:names:tc:ebcore:partyid-type:unregistered:receiver/services/::eftigateaction`.

Jõudlustestide tulemusi kirjeldab tabel 6.

Tabel 6. Harmony SMP jõudlustestide tulemused

| Päringute arv (-n) | Samaaegsed päringud (-c) | Päringuid sekundis (mediaan) | 99% latentsus (ms) |
|---|---|---|---|
| 100 | 10 | 132 | 75 |
| 1000 | 100 | 178 | 559 |
| 5000 | 250 | 209 | 1191 |
| 10000 | 1000 | 211 | 4725 |

Võrreldes kõigi teiste SMP lahendustega on Harmony SMP koormustaluvuse mõttes kõige kehvem. Päringute arv sekundis on piiratud kuid suurema koormuse puhul tõuseb vastuste latentsus.

## 4.5 Järeldused

SMP serverite jõudlustestide tulemusi saab võrrelda kasutades tabelit 7. Punasega on tähistatud kõige halvem tulemus. Rohelisega kõige parem tulemus.

Kasutades minimaalseid tööriistu ning luues tarkvara, mis täidab kindlat ülesannet on võimalik luua väga kiire lahendus.

Tabel 7. SMP jõudlustestide võrdlus

| Koormus | | Päringuid sekundis (mediaan) | | |
|---|---|---|---|---|
| Päringute arv (-n) | Samaaegsed päringud (-c) | POC SMP | Phoss SMP | Harmony SMP |
| 100 | 10 | 3134 | 1245 | 132 |
| 1000 | 100 | 920 | 3080 | 178 |
| 5000 | 250 | 3220 | 1297 | 209 |
| 10000 | 1000 | 4920 | 1290 | 211 |
| Koormus | | 99% latentsus (ms) | | |
| Päringute arv (-n) | Samaaegsed päringud (-c) | POC SMP | Phoss SMP | Harmony SMP |
| 100 | 10 | 3.2 | 8 | 75 |
| 1000 | 100 | 108 | 32 | 559 |
| 5000 | 250 | 77 | 192 | 1191 |
| 10000 | 1000 | 200 | 774 | 4725 |

# Kasutatud kirjandus


[1] Euroopa Liit. *Euroopa ühendamise rahastu (CEF)*. EUR-Lex kokkuvõtted. Accessed: 08-02-2026. 2026. URL: https://eur-lex.europa.eu/ET/legal-content/glossary/connecting-europe-facility-cef.html.

[2] European Commission. *EFTI4EU: Electronic Freight Transport Information for Europe (Project No. 101122891)*. EU Funding & Tenders Portal. Accessed: 08-02-2026. 2023. URL: https://ec.europa.eu/info/funding-tenders/opportunities/portal/screen/opportunities/projects-details/43251567/101122891.

[3] EFTI4EU Consortium. *EFTI4EU Reference Implementation*. GitHub repository. Accessed: 08-02-2026. 2025. URL: https://github.com/EFTI4EU/reference-implementation/.

[4] Euroopa Parlament ja Euroopa Liidu Nõukogu. *Komisjoni delegeeritud määrus (EL) 2024/2024, 26. juuli 2024, millega täiendatakse määrust (EL) 2020/1056, kehtestades eFTI ühise andmekomplekti ja andmealamhulgad*. Euroopa Liidu Teataja, L 2024/2024, 20. detsember 2024. Vaadatud: 08-02-2026. 2024. URL: https://eur-lex.europa.eu/legal-content/ET/TXT/?uri=CELEX:32024R2024.

[5] Ulrika Hurt, Christian Lüpges ja Tom Annikve. *Analysis of the eFTI Regulation 2020/1056 Requirements for eFTI Platforms, Service Providers and Data Exchange with a Focus on Requirements on eFTI Platform on Electronic Road Transport Waybills (eCMRs)*. Digilogistika Keskus OÜ / Digital Logistics Centre of Excellence. Significant input: Heiti Mering; Originally compiled in Estonian; Vaadatud: 08-02-2026. Estonia, apr 2022. URL: https://realtimeeconomy-bsr.eu/sites/global/files/2022-08/REPORT_eFTI%20requirements%20analysis%20for%20eFTI%20platforms%20Estonia%20_April%202022%20%28FINAL%29.pdf.

[6] Riigi Infosüsteemi Amet. *eDelivery selgitus*. RIA abiportaal. Vaadatud: 08-02-2026. 2026. URL: https://abi.ria.ee/sdg/edelivery-selgitus.

[7] *X-tee*. Vikipeedia, vaba entsüklopeedia. Vaadatud: 01-03-2026. 2026. URL: https://et.wikipedia.org/wiki/X-tee.

[8] Petteri Kivimäki. *X-Road and eDelivery – Identical Twins or Distant Relatives?* NIIS blog. Vaadatud: 01-03-2026. 2019. URL: https://www.niis.org/blog/2019/9/26/x-road-and-edelivery-identical-twins-or-distant-relatives.

[9] Euroopa Parlament ja Euroopa Liidu Nõukogu. *Komisjoni rakendusmäärus (EL) 2024/1942, 5. juuli 2024, millega kehtestatakse ühtsed menetlused ja üksikasjalikud normid seoses pädevate asutuste juurdepääsuga elektroonilisele kaubaveoteabele ja selle töötlemisega kooskõlas Euroopa Parlamendi ja nõukogu määrusega (EL) 2020/1056*. Euroopa Liidu

Teataja, L 1942, 20. detsember 2024. Vaadatud: 08-02-2026. 2024. URL: https://eur-lex.europa.eu/legal-content/ET/TXT/?uri=CELEX:32024R1942.

[10] *The eFTI Regulation*. European Commission – Mobility and Transport. Vaadatud: 01-03-2026. 2026. URL: https://transport.ec.europa.eu/transport-themes/logistics-and-multimodal-transport/efti-regulation_en.

[11] European Commission. *eDelivery*. European Commission – Digital Building Blocks. Vaadatud: 29-03-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/467110114/eDelivery.

[12] European Commission. *eDelivery AS4 Conformant Solutions*. European Commission – Digital Building Blocks. Vaadatud: 29-03-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/467110150/eDelivery+AS4+conformant+solutions.

[13] European Commission. *eDelivery AS4 v1.x Conformant Products*. European Commission – Digital Building Blocks. Vaadatud: 29-03-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/721846393/eDelivery+AS4+v1.x+conformant+products.

[14] European Commission. *Domibus – eDelivery Sample AS4 Implementation (Repository)*. European Commission – Digital Building Blocks Code. Vaadatud: 29-03-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/domibus/browse.

[15] Mattias Malk. *eFTI platvormide ja riiklike juurdepääsupunktide-vaheline andmevahetus*. Tallinna Tehnikaülikool – Digikogu. Bakalaureusetöö. Vaadatud: 04-04-2026. 2024. URL: https://digikogu.taltech.ee/et/item/017d8c9f-b179-46ca-85fc-570a9842b937.

[16] Riigi Infosüsteemi Amet. *Harmony Access Point*. RIA – SDG (Single Digital Gateway) dokumentatsioon. Vaadatud: 04-04-2026. 2026. URL: https://abi.ria.ee/sdg/harmony-access-point.

[17] Nordic Institute for Interoperability Solutions. *harmony-access-point: Harmony eDelivery Access – Access Point*. GitHub repository. Vaadatud: 04-04-2026. 2025. URL: https://github.com/nordic-institute/harmony-access-point.

[18] Nordic Institute for Interoperability Solutions. *harmony-common: Harmony eDelivery Access - Common repository including documentation and packaging scripts*. GitHub repository. Vaadatud: 04-04-2026. 2025. URL: https://github.com/nordic%E2%80%91institute/harmony%E2%80%91common.

[19] Nordic Institute for Interoperability Solutions. *niis/harmony-ap: Containerized Harmony eDelivery Access Point*. Docker Hub container image. Vaadatud: 04-04-2026. 2026. URL: https://hub.docker.com/r/niis/harmony-ap.

[20] phax (Philip Helger). *phase4: Lightweight AS4 client and server with built-in support for Peppol and CEF eDelivery*. GitHub repository. Vaadatud: 04-04-2026. 2025. URL: https://github.com/phax/phase4.

[21] Philip Helger. *Philip Helger – LinkedIn Profile*. LinkedIn. Vaadatud: 04-04-2026. 2026. URL: https://www.linkedin.com/in/void0/.

[22] European Commission. *eDelivery SMP Conformant Solutions*. European Commission – Digital Building Blocks. Vaadatud: 04-04-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/467117765/eDelivery+SMP+conformant+solutions.

[23] European Commission. *eDelivery SMP - 2.0*. European Commission – Digital Building Blocks. Vaadatud: 04-04-2026. 2025. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/843612543/eDelivery%2BSMP%2B-%2B2.0.

[24] Philip Helger / phax. *phoss-SMP (Service Metadata Publisher Implementation)*. GitHub repository. Vaadatud: 04-04-2026. 2026. URL: https://github.com/phax/phoss-smp.

[25] phelger / Docker Hub. *phelger/phoss-smp-xml Docker Image Tags*. Docker Hub repository. Vaadatud: 04-04-2026. 2026. URL: https://hub.docker.com/r/phelger/phoss-smp-xml/tags.

[26] Nordic Institute for Interoperability Solutions. *Harmony eDelivery Access – Service Metadata Publisher (SMP)*. GitHub repository. Vaadatud: 05-04-2026. 2026. URL: https://github.com/nordic-institute/harmony-smp.

[27] Nordic Institute for Interoperability Solutions. *eDelivery Components*. Harmony eDelivery Access documentation. Vaadatud: 05-04-2026. 2026. URL: https://edelivery.digital/components.

[28] European Commission. *eDelivery SMP Repository*. European Commission – Digital Building Blocks Code. Vaadatud: 05-04-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/smp/browse.

[29] European Commission. *eDelivery Service Metadata Locator (SML) service*. European Commission – Digital Building Blocks. Vaadatud: 05-04-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/467118484/SML+service.

[30] European Commission. *DomiSML FAQ*. European Commission – Digital Building Blocks. Vaadatud: 05-04-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/850362424/DomiSML+FAQ.

[31] European Commission. *BDMSL (Service Metadata Locator) Repository*. European Commission – Digital Building Blocks Code. Vaadatud: 05-04-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/code/projects/EDELIVERY/repos/bdmsl/browse.

[32] Ademico Software. *Domibus REST API Plugin*. Ademico Software Website. Vaadatud: 23-04-2026. 2026. URL: https://ademico-software.com/domibus-rest-api-plugin/.

[33] Tanel Tammet. „Veebiteenused reeglite ikkes“. *Arvutimaailm* 2 (2007). Vaadatud: 23-04-2026. Kättesaadav ka: https://artiklid.elnet.ee/record=b1020192*est, lk-d 22–24. URL: http://lambda.ee/wiki/Veebiteenuste_v%C3%B5lu_ja_valu.

[34] Wireshark Foundation. *Wireshark · Go Deep*. Wireshark Official Website. Vaadatud: 26-04-2026. 2026. URL: https://www.wireshark.org/.

[35] European Commission. *Domibus Documentation - Release 5.2*. Vaadatud: 26-04-2026. 2026. URL: https://docs.edelivery.tech.ec.europa.eu/domibus/5.2/domibus-documentation-5.2.pdf.

[36] Zone Media OÜ. *CNAME kirje*. Zone Help – Tehniline abiinfo. Vaadatud: 03-05-2026. Viimati uuendatud: 18.02.2026. 2026. URL: https://www.zone.ee/help/kb/nimeserveri-cname-kirje/.

[37] OpenPeppol AISBL. *Peppol CNAME to NAPTR Migration Process v1.0.0*. Vaadatud: 23-04-2026. Last updated: 17.04.2025. 2025. URL: https://docs.peppol.eu/edelivery/changelog/2025-04/Peppol%20CNAME%20to%20NAPTR%20Migration%20Process%20v1.0.0%202025-04-17.pdf.

[38] Michael Mealling. *Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database*. RFC 3403. Vaadatud: 03-05-2026. Standard Track. Network Working Group, okt 2002. URL: https://datatracker.ietf.org/doc/html/rfc3403.

[39] Ionite. *Do we need an SML?* Ionite - News & Articles. Vaadatud: 03-05-2026. Postitatud: 27.02.2024. 2024. URL: https://ionite.net/news-articles/2024-02-27_do_we_need_an_sml/.

[40] Keksworks. *K-Lite: A lightweight Kotlin library for web services*. GitHub Repository. Vaadatud: 03-05-2026. 2026. URL: https://github.com/keksworks/klite/blob/main/README.md.

[41] Brian Wellington ja Ingo Bauersachs. *dnsjava: A DNS implementation in Java*. GitHub Repository. Vaadatud: 03-05-2026. 2026. URL: https://github.com/dnsjava/dnsjava.

[42] Kenneth Bengtsson ja Erlend Klakegg Bergheim. *Service Metadata Publishing (SMP) Version 2.0*. Vaadatud: 05-05-2026. OASIS Standard, 14 May 2021. 2021. URL: https://docs.oasis-open.org/bdxr/bdx-smp/v2.0/bdx-smp-v2.0.html.

[43] European Commission. *ebCore Party ID specifications*. Vaadatud: 04-05-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/467118569/ebCore+Party+ID+specifications.

[44] Dale Moberg ja Pim van der Eijk. *OASIS ebCore Party Id Type Technical Specification Version 1.0*. Vaadatud: 04-05-2026. OASIS Standard, 28 September 2010. 2010. URL: https://docs.oasis-open.org/ebcore/PartyIdType/v1.0/PartyIdType-1.0.html.

[45] European Commission. *eDelivery SMP - 2.1*. Vaadatud: 05-05-2026. 2026. URL: https://ec.europa.eu/digital-building-blocks/sites/spaces/DIGITAL/pages/930451568/eDelivery+SMP+-+2.1.

[46] European Commission. *Disposition of Public Review Comments on eDelivery SMP 2.0 - 2024 PR draft v1.0*. Vaadatud: 05-05-2026. 2024. URL: https://ec.europa.eu/digital-building-blocks/sites/download/attachments/711494220/%28eDelivery%29.%28Disposition%20of%20Public%20Review%20Comments%20on%20eDelivery%20SMP%202.0%20-%202024%20PR%20draft%29.%28v1.0%29.pdf?api=v2.

[47] OpenPeppol AISBL. *Peppol AS4 Profile Technical Specification v2.0.2*. Vaadatud: 07-05-2026. 2024. URL: https://docs.peppol.eu/edelivery/as4/specification/.

[48] Rig14. *Harmony SMP Dockerized: A containerized Peppol Service Metadata Publisher*. GitHub Repository. Vaadatud: 07-05-2026. 2026. URL: https://github.com/Rig14/harmony-smp-dockerized.

[49] Rig14. *harmony-smp-mysql: Docker image for Harmony SMP with MySQL backend*. Docker Hub Repository. Vaadatud: 07-05-2026. 2026. URL: https://hub.docker.com/repository/docker/rig14/harmony-smp-mysql/general.

[50] Mark Bartel *et al. XML Signature Syntax and Processing Version 1.1*. Vaadatud: 07-05-2026. W3C Recommendation 11 April 2013. 2013. URL: https://www.w3.org/TR/xmldsig-core1/.

[51] XimpleWare. *XimpleWare Benchmark Report*. Vaadatud: 2026-05-07. 2003. URL: http://ximpleware.com/old/benchmark1.html.

[52] John Boyer ja Glenn Marcy. *XML Canonicalization (C14N) Version 1.1*. Vaadatud: 07-05-2026. W3C Recommendation 2 May 2008. 2008. URL: https://www.w3.org/TR/xml-c14n11/#XMLCanonicalization.

[53] John Boyer. *Canonical XML Version 1.0*. Vaadatud: 07-05-2026. W3C Recommendation 15 March 2001. 2001. URL: https://www.w3.org/TR/2001/REC-xml-c14n-20010315.

[54] Donald Eastlake, Joseph Reagle ja David Solo. *XML-Signature Syntax and Processing*. RFC 3275. Vaadatud: 07-05-2026. Standard Track. Internet Engineering Task Force, märts 2002. URL: https://datatracker.ietf.org/doc/html/rfc3275.

[55] Apache Software Foundation. *ab - Apache HTTP server benchmarking tool*. Vaadatud: 09-05-2026. 2026. URL: https://httpd.apache.org/docs/2.4/programs/ab.html.

[56] DNS-OARC. *dnsperf and resperf: DNS Performance Testing Tools*. Codeberg Repository. Vaadatud: 09-05-2026. 2026. URL: https://codeberg.org/DNS-OARC/dnsperf.

# Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks[1]

Mina, Rihard Rivis

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose "eDelivery SML ja SMP kasutamine eFTI süsteemis", mille juhendajad on Ulrika Hurt ja Tarvo Treier
    1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
    1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

10.05.2026

[1] Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingulise tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

# Lisa 2 – Käsklused

Käsklus 1. Koodibaasi ridade arvu leidmise käsklus

```
git ls-files -z | xargs -0 cat | wc -l
```

Käsklus 2. OpenSSL käsklus ise allkirjastatud sertifikaatide ja privaatvõtmete loomiseks

```
openssl genpkey -algorithm RSA -out "$cert_dir/$NAME.key" -pkeyopt
   rsa_keygen_bits:2048
openssl req -new -x509 -key "$cert_dir/$NAME.key" -out
   "$cert_dir/$NAME.crt" -days 1825 -subj "/CN=$party_id"
openssl pkcs12 -export -out "$cert_dir/$NAME.p12" -inkey
   "$cert_dir/$NAME.key" -in "$cert_dir/$NAME.crt" -name
   "$party_id" -passout pass:changeit
```

Käsklus 3. ab tööriista kasutamise käsklus SMP serveri jõudluse hindamiseks

```
SERVER_URL = http://localhost:8080/urn%3Aoasis%3Anames%3Atc%3Aebcore%3Apartyid-
type%3Aunregistered%3A%3Areceiver/services/%3A%3AeftiGateAction
ab -n $TOTAL_REQUESTS -c $CONCURRENCY $SERVER_URL
```

Käsklus 4. dnsperf tööriista kasutamise käsklus SML serveri jõudluse hindamiseks

```
dnsperf -s 127.0.0.1 -p 8053 -d <(yes "
SG7WQEA6Z27EYWS575WTM3AZ6SVFZIY7G6TRTKZDWC45AZRV6OEQ.testsml NAPTR" | head -n
$TOTAL_REQUESTS) -c $CONCURRENCY
```

# Lisa 3 – PMode sender konfiguratsioon

Koodinäidis 5. Juurdepääsupunkti sender PMode konfiguratsioon

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<domibus:configuration xmlns:domibus="http://domibus.eu/configuration" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://domibus.eu
/configuration␣../../../schema/pmode/configuration.xsd" party="sender">
    <businessProcesses>
        <roles>
            <role name="defaultInitiatorRole"
                  value="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/2
00704/initiator"/>
            <role name="defaultResponderRole"
                  value="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/2
00704/responder"/>
        </roles>
        <parties>
            <partyIdTypes>
                <partyIdType name="partyTypeUrn" value="urn:oasis:names:tc:ebcore
:partyid-type:unregistered"/>
                <partyIdType name="urn:oasis:names:tc:ebcore:partyid-type:EU-LU3"
 value=""/>
            </partyIdTypes>
            <party name="sender" endpoint="http://harmony-ap-sender-1:8080/servic
es/msh">
                <identifier partyId="sender" partyIdType="partyTypeUrn"/>
            </party>
        </parties>
        <meps>
            <mep name="oneway" value="http://docs.oasis-open.org/ebxml-msg/ebms/v
3.0/ns/core/200704/oneWay" legs="0"/>
            <binding name="push" value="http://docs.oasis-open.org/ebxml-msg/ebms
/v3.0/ns/core/200704/push"/>
        </meps>
        <properties>
```

```
            <property name="originalSenderProperty" key="originalSender" datatype
="string" required="true"/>
            <property name="finalRecipientProperty" key="finalRecipient" datatype
="string" required="true"/>
            <propertySet name="eDeliveryPropertySet">
                <propertyRef property="finalRecipientProperty"/>
                <propertyRef property="originalSenderProperty"/>
            </propertySet>
        </properties>
        <payloadProfiles>
            <payload name="businessContentPayload" cid="cid:message" mimeType="te
xt/xml" maxSize="0" required="true"
                     inBody="false"/>
            <payload name="businessContentAttachment" cid="cid:attachment" mimeTy
pe="application/octet-stream"
                     maxSize="0"
                     required="false" inBody="false"/>
            <payloadProfile name="MessageProfile" maxSize="40894464">
                <attachment name="businessContentPayload"/>
                <attachment name="businessContentAttachment"/>
            </payloadProfile>
        </payloadProfiles>
        <securities>
            <security name="eDeliveryAS4Policy" policy="eDeliveryAS4Policy.xml" s
ignatureMethod="RSA_SHA256"/>
        </securities>
        <errorHandlings>
            <errorHandling name="errorHandling" errorAsResponse="true" businessEr
rorNotifyProducer="true"
                           businessErrorNotifyConsumer="true" deliveryFailureNoti
fyProducer="true"/>
        </errorHandlings>
        <agreements>
            <agreement name="agreement1" value="A1" type="T1"/>
        </agreements>
        <services>
            <service name="eftiGateService" value="bdx:noprocess" type="tc1"/>
            <service name="pingService"
```

```
                    value="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/cor
e/200704/service"/>
        </services>
        <actions>
            <action name="eftiGateAction" value="eftiGateAction"/>
            <action name="pingAction"
                    value="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core
/200704/test"/>
        </actions>
        <as4>
            <receptionAwareness name="receptionAwareness" retry="1;5;CONSTANT" du
plicateDetection="true"/>
            <reliability name="AS4Reliability" replyPattern="response" nonRepudia
tion="true"/>
        </as4>
        <legConfigurations>
            <legConfiguration name="eftiGateLeg"
                              service="eftiGateService"
                              action="eftiGateAction"
                              defaultMpc="defaultMpc"
                              reliability="AS4Reliability"
                              security="eDeliveryAS4Policy"
                              receptionAwareness="receptionAwareness"
                              propertySet="eDeliveryPropertySet"
                              payloadProfile="MessageProfile"
                              errorHandling="errorHandling"
                              compressPayloads="true"/>
            <legConfiguration name="pingServiceCase"
                              service="pingService"
                              action="pingAction"
                              defaultMpc="defaultMpc"
                              reliability="AS4Reliability"
                              security="eDeliveryAS4Policy"
                              receptionAwareness="receptionAwareness"
                              propertySet="eDeliveryPropertySet"
                              payloadProfile="MessageProfile"
                              errorHandling="errorHandling"
                              compressPayloads="true"/>
        </legConfigurations>
```

```
        <process name="eftiProcess" initiatorRole="defaultInitiatorRole" responde
rRole="defaultResponderRole"
                 mep="oneway"
                 binding="push">
            <initiatorParties>
                <initiatorParty name="sender"/>
            </initiatorParties>
            <legs>
                <leg name="eftiGateLeg"/>
                <leg name="pingServiceCase"/>
            </legs>
        </process>
    </businessProcesses>
    <mpcs>
        <mpc name="defaultMpc" retention_downloaded="0" retention_undownloaded="1
4400" retention_sent="-1"
             retention_metadata_offset="0" delete_message_metadata="false" max_ba
tch_delete="-1" default="true"
             enabled="true" qualifiedName="http://docs.oasis-open.org/ebxml-msg/e
bms/v3.0/ns/core/200704/defaultMPC"/>
    </mpcs>
</domibus:configuration>
```

# Lisa 4 – WS Plugin näidissõnum

Koodinäidis 6. WSPlugin näidissõnum

```
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope" xmlns:env="http://w
ww.w3.org/2003/05/soap-envelope">
    <S:Header>
        <ns5:Messaging xmlns:ns3="http://eu.domibus.wsplugin/" xmlns:ns5="http://
docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/" xmlns:xmime="http://www.
w3.org/2005/05/xmlmime">
            <ns5:UserMessage>
                <ns5:MessageInfo>
                    <ns5:MessageId>b840a46c-e3e7-11f0-9888-5a2df3995194@domibus.e
u</ns5:MessageId>
                </ns5:MessageInfo>
                <ns5:PartyInfo>
                    <ns5:From>
                        <ns5:PartyId type="urn:oasis:names:tc:ebcore:partyid-type
:unregistered">estonia</ns5:PartyId>
                        <ns5:Role>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/
ns/core/200704/initiator</ns5:Role>
                    </ns5:From>
                    <ns5:To>
                        <ns5:PartyId type="urn:oasis:names:tc:ebcore:partyid-type
:unregistered">borduria</ns5:PartyId>
                        <ns5:Role>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/
ns/core/200704/responder</ns5:Role>
                    </ns5:To>
                </ns5:PartyInfo>
                <ns5:CollaborationInfo>
                    <ns5:Service type="tc1">bdx:noprocess</ns5:Service>
                    <ns5:Action>eftiGateAction</ns5:Action>
                    <ns5:ConversationId>63ac78ba-60f8-4f04-9f57-e3cd8ab2ca9c</ns5
:ConversationId>
                </ns5:CollaborationInfo>
                <ns5:MessageProperties>
```

```
                    <ns5:Property name="originalSender">urn:oasis:names:tc:ebcore
:partyid-type:unregistered:C1</ns5:Property>
                    <ns5:Property name="finalRecipient">urn:oasis:names:tc:ebcore
:partyid-type:unregistered:C4</ns5:Property>
                </ns5:MessageProperties>
                <ns5:PayloadInfo>
                    <ns5:PartInfo href="cid:message">
                        <ns5:PartProperties>
                            <ns5:Property name="MimeType">text/xml</ns5:Property>
                        </ns5:PartProperties>
                    </ns5:PartInfo>
                </ns5:PayloadInfo>
                <ns5:ProcessingType>PUSH</ns5:ProcessingType>
            </ns5:UserMessage>
        </ns5:Messaging>
    </S:Header>
    <S:Body>
        <ns3:submitRequest xmlns:ns3="http://eu.domibus.wsplugin/" xmlns:ns5="htt
p://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/" xmlns:xmime="http://
www.w3.org/2005/05/xmlmime">
            <payload contentType="text/xml" payloadId="cid:message">
                <value>PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiIHN0YW5kYW
xvbmU9InllcyI/Pjx1aWxRdWVyeSByZXF1ZXN0SWQ9IjYzYWM3OGJhLTYwZjgtNGYwNC05ZjU3LWUzY2Q
4YWIyY2E5YyIgeG1sbnM9Imh0dHA6Ly9lZnRpLmV1L3YxL2VkZWxpdmVyeSIgeG1sbnM6bnMyPSJodHRw
Oi8vZWZ0aS5ldS92MS9jb25zaWdubWVudC9pZGVudGlmaWVyIj48dWlsPjxnYXRlSWQ+Ym9yZHVyaWE8L
2dhdGVJZD48cGxhdGZvcm1JZD5wbGF0MTwvcGxhdGZvcm1JZD48ZGF0YXNldElkPjUyODZhZWU4LWE1ZT
UtNDQ5OS1hMWEwLTQwOWY3MzQzZGZlYzwvZGF0YXNldElkPjwvdWlsPjxzdWJzZXRJZD5FVTAxPC9zdWJ
zZXRJZD48L3VpbFF1ZXJ5Pg==</value>
            </payload>
        </ns3:submitRequest>
    </S:Body>
</S:Envelope>
```

# Lisa 5 – Sõnumi saaja metaandmed

Koodinäidis 7. SMP serveri poolt genereeritud sõnumi saaja metaandmed


```
<?xml version="1.0" encoding="UTF-8"?>
<SignedServiceMetadata xmlns="http://docs.oasis-open.org/bdxr/ns/SMP/2016/05">
    <ServiceMetadata>
        <ServiceInformation>
            <ParticipantIdentifier>receiver</ParticipantIdentifier>
            <DocumentIdentifier>eftiGateAction</DocumentIdentifier>
            <ProcessList>
                <Process>
                    <ProcessIdentifier scheme="tc1">bdx:noprocess</ProcessIdentif
ier>
                    <ServiceEndpointList>
                        <Endpoint transportProfile="bdxr-transport-ebms3-as4-v1p0
">
                            <EndpointURI>http://ap-receiver:8080/services/msh</En
dpointURI>
                            <Certificate>
                                MIIDBzCCAe+gAwIBAgIUbuRbzUbZmORNjZar6OrGHZf3f54wD
QYJKoZIhvcNAQELBQAwEzERMA8GA1UEAwwIcmVjZWl2ZXIwHhcNMjYwMTIzMTM1NzI1WhcNMzEwMTIyMT
M1NzI1WjATMREwDwYDVQQDDAhyZWNlaXZlcjCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMl
4UeYRkI9eC958NgYsBufFFtgg8WjL+DeFSUkCfY+sK78yo0fQygxrVbktw4wJVLJ/aj8vLt/fDFXYdoGN
hv5pVT1qlUyGCG0+jVK66Z+WjR3PuLAZ24XPDacTqCe1tbmzRItmUwQWIV+HIXxChGYu0/S7Cz4xgIP8Q
fczO0FBrZIS88+Ku1LGrv6hRjAlrvz2RVL7mk1rhxdLeNy3aKVhXeifA/sQo3YDdGPtvNCO+ICjxIuROh
j3YgZ9Y1Dl8H978j6GHtZGFK4k02ol45MklgsRD7mKgvZraS25ZwTpSWMp+dw2pdU6L5Gb7Vr+wWeGQ1j
b/97TshqVx7Te230CAwEAAaNTMFEwHQYDVR0OBBYEFOSf7RTt28wd1GBaOqcYdgnWK7AqMB8GA1UdIwQY
MBaAFOSf7RTt28wd1GBaOqcYdgnWK7AqMA8GA1UdEwEB/wQFMAMBAf8wDQYJKoZIhvcNAQELBQADggEBA
F3WWPoK8BghNw6Ihai6aX2xv2xrF+Vo+XT6ErEg0jnQVk3LGFlSbNaAjevhPp/8+VUROqrf52zto219+r
wkjbPs7/ApTS/UabmUjys9vv43Mla2aLq/iqlk4QZQf6g/dmFOrzTnFosvLrruWUBCT8g7Er8hqYHwmit
SUOiLupAIDSLqGtffxQKnao7M5PxP0ihXyoXVzlQ7kSqqydYp3WZFsfR9GIkgmRk5tGb3bR18EMdcPKDh
ZFXJrNEG6Xz4fmY+zT1stpsdhvUMADvh5j4+76vCHSf/EjAWjz4A+n6S6u4QR/s03EHb8jhGEJxr429Lz
AowRgiz/dveLmPxu3A=</Certificate>
                            <ServiceDescription>This service is used to upload id
entifiers to the
```

```
                                eFTI Gate</ServiceDescription>
                            <TechnicalContactUrl>contact@efti.eu</TechnicalContac
tUrl>
                        </Endpoint>
                    </ServiceEndpointList>
                </Process>
            </ProcessList>
        </ServiceInformation>
    </ServiceMetadata>
    <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
        <SignedInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
            <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-
c14n-20010315"></CanonicalizationMethod>
            <SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rs
a-sha256"></SignatureMethod>
            <Reference URI="">
                <Transforms>
                    <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#envel
oped-signature"></Transform>
                </Transforms>
                <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
></DigestMethod>
                <DigestValue>s1sO+7+OiA+eAXMyiSaL8fOuPh9koFtIb+1U34nvO1U=</Digest
Value>
            </Reference>
        </SignedInfo>
        <SignatureValue>
            AFJv9g3qa4ObL5GQRlWjm79FxnduQF1TjYcjLFKELstebGYS6i2scihsIAgGAD6fNk30X
nomCKfHsbJ4VIUqIWzefU824XeXvTsh0tZ1FxQrBKg6gD2Rrb+4J3PKuyWu3Y8+zd6oS910KESBaKN801
u2+OAJpn/HoBGRSbPmei+yEWCgyW4rn1+lk8/2yie/zP6zQzms8vqNob24FCrmhmRE5XWdjHKgo6HFIfI
W7lg4cWTwvLP29FkigqxpinNm2Vkh1+nv1ZcPmIOtuq9ir4Ta68VtfEzduJ32DE9ohd7pvPTVH6EY8073
DPtwL2qtilCN8/IcAFhcA17UWqOMYA==</SignatureValue>
        <KeyInfo>
            <X509Data>
                <X509SubjectName>CN=smp</X509SubjectName>
                <X509Certificate>
                    MIIC/TCCAeWgAwIBAgIUCDwqXg/F9gY4fgzPSCZePs5ZnXwwDQYJKoZIhvcNA
QELBQAwDjEMMAoGA1UEAwwDc21wMB4XDTI2MDUwNzEwMDM0OVoXDTMxMDUwNjEwMDM0OVowDjEMMAoGA1
UEAwwDc21wMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsWf1hrqIU14eccD1LqNpRe7dFhO
```

FXK55rzssy3QaBKhta4u8f0R0B1POmmkIdaIy7V/4wZEC5ah+7KXnKzStE7UVLYpXCdh5FBK2LvYNwi77
TPLnC3eZijOde9O/LpGQjEEsMeaBJPfAKaRpGQiiuykKnX937sBCJ/ssXLviz/ow7ShkERecQHQKJps+W
qLbkpA8swBHsc1AUYCPrW3AOWFhFyOlVm29Tmxt+MuvmgfXYjp6cnTfD/xwSTBJEgGoxChpW25qfiffl2
jL+Ogq4Cq4JVlcH3zehweLeVsZ/UBfQ/BPID+xqZyJaJOOrRSWibBECiEUjHUjZMNeOmLOZQIDAQABo1M
wUTAdBgNVHQ4EFgQUIzdg2vbeejpheJ1u3+CLcuIpPBIwHwYDVR0jBBgwFoAUIzdg2vbeejpheJ1u3+CL
cuIpPBIwDwYDVR0TAQH/BAUwAwEB/zANBgkqhkiG9w0BAQsFAAOCAQEAV2aSqgRuqsZ45hn5I83dI1Mog
sosPrnpTIeDVqzSOtCH3lC7FXn7aAWKOKQZgSqXXMbPTTUDJWB2SbA7gLHy/4zJal+8aYaRGPiAvDtb7a
ccu5sPByiinw5p9wGmxvOZWbSbZsT1dxGxYqEaBo6sM5pZnoaXrlLiRVefWdsdrKW1o2SqV8ebt3IETLd
yZKLhkd/8nnqvACSnWOQkpxQ6zgdE2KLPagNkOKpgNCi4nXdeDlQVNDc2Zwzgayn4qTCG4d9PG3rtcZNH
8uvDRnGGlG1ANTQOjtlv/P43NYp9Ci5ZSs+WYUe6itCqW/2pjMLvP+lV2HlElR9K2zxHM3ujpQ==</X50
9Certificate>
            </X509Data>
        </KeyInfo>
    </Signature>
</SignedServiceMetadata>