IVXV protokollid

Spetsifikatsioon

Versioon 1.0

20. september 2017

50 lk

Dok IVXV-PR-1.0

Sisukord

PEATÜKK 1

Annotatsioon

Käesolev dokument kirjeldab elektroonilise hääletamise infosüsteemi IVXV protokollistikku.

Dokument annab üldise ülevaate elektroonilise hääletamise süsteemi tehnilisest ülesehitusest ja kasutatavatest protokollidest. Dokumendis defineeritakse protokollides kasutatavad ühised mõisted ja andmestruktuurid.

PEATÜKK 2

Ülevaade

Elektroonilise hääletamise protokollistik (edaspidi protokollistik) defineerib elektroonilise hääletamise süsteemi komponentide vahelise sõnumivahetuse, kasutatavad andmestruktuurid, algoritmid ning liidesed väliste süsteemidega. Sõnumivahetus esitatakse UML suhtlusskeemidena, mis üheselt defineerivad sõnumite järgnevuse. Andmestruktuuride kirjeldused on varustatud Backus-Naur või JSON-schema notatsioonis spetsifikatsioonidega. Andmestruktuuride väljade eraldajateks kasutatakse reavahetuse sümbolit LF, ASCII-koodiga 0x0A ja tabulaatori sümbolit TAB, ASCII-koodiga 0x09. Algoritmid esitatakse pseudokoodina.

NB! Kõigis protokollistiku andmestruktuuride väljades tuleb rangelt kinni pidada lubatud sümbolitest ning väljade minimaalsetest-maksimaalsetest pikkustest. Täiendavate tühikute, tabulaatorite jms. kasutamine on keelatud ning spetsifikatsiooni realiseerivad rakendused peavad formaadile mitte-vastavate andmete töötlemisest keelduma.

Protokollistik defineerib elektroonilise hääletamise protokolli ning selle protokolli realiseerimiseks vajalikud tugistruktuurid.

2.1 Elektroonilise hääletamise protokoll

Elektroonilise hääletamise protokoll spetsifitseerib

1. elektroonilise hääle formaadi, mis võimaldab üheselt määratleda valija tahte konkreetsel valimisel;
2. elektroonilise hääle krüpteerimise hääle salajasuse tagamiseks;
3. elektroonilise hääle digitaalse allkirjastamise tervikluse ja valija identifitseerimise tagamiseks;
4. elektroonilise hääle kvalifitseerimise kogumisteenuse poolt, hääle vastu võtmise tähistamiseks;

Protokoll eeldab, et valimise korraldaja on defineerinud valimise ning genereerinud häälte salastamise võtmepaari, mille avalik komponent on tehtud valijarakendusele kättesaadavaks.

Protokolli vahendusel liigub valija tahe kogumisteenuses talletatavasse e-urni ning võetakse tulemuse kujunemisel arvesse järgmist sündmusterida pidi:

1. Valija kasutab valijarakendust oma tahteavalduse vormistamiseks elektrooniliselt
   1.1 tahteavaldus vormistatakse elektroonilise häälena,
   1.2 vormistatud hääl krüpteeritakse,
   1.3 krüpteeritud hääl allkirjastatakse digitaalselt.
2. Kogumisteenus talletab elektroonilise hääle
   2.1 digitaalselt allkirjastatud häälele võetakse valija sertifikaadi kehtivust kinnitavad elemendid,
   2.2 elektrooniline hääl registreeritakse välises registreerimisteenuses,
   2.3 valijale võimaldatakse kvalifitseeritud elektroonilise hääle kontrollimine kontrollrakendusega.
3. Valija võib kasutada kontrollrakendust veendumaks oma hääle korrektses käitlemises kogumisteenuse poolt.
4. Hääletamisperioodi lõppedes väljastab kogumisteenus valimise korraldajale e-urni ning registreerimisteenus loendi kogumisteenuse poolt registreeritud häältest.
5. Valimise korraldaja arvutab hääletamistulemuse
   5.1 veendutakse, et kõik registreerimisteenuses registreeritud hääled on e-urni koosseisus üle antud
   5.2 eraldatakse krüpteeritud hääled ja digitaalallkirjad,
   5.3 dekrüpteeritakse krüpteeritud hääled,
   5.4 dekrüpteeritud häälte põhjal arvutatakse hääletamistulemus.

Protokoll on analoogne paberil posti teel hääletamise protokolliga, kus valija tahe liigub valimiskomisjonini kahes ümbrikus – välimise ümbriku sees on sisemine ümbrik, mis omakorda sisaldab valija tahteavaldusega hääletussedelit. Välimine ümbrik kannab valijat identifitseerivat informatsiooni ning võimaldab mh. kontrollida valija õigust hääletada. Sisemine ümbrik on anonüümne ning kaitseb hääle salajasust. Enne häälte kokkulugemist eraldatakse sisemised ümbrikud välimistest.

Elektroonilise hääletamise kontekstis on sisemine ümbrik vormistatud krüpteeritud häälena ning välimine ümbrik digitaalselt allkirjastatud dokumendina.

PEATÜKK 3

Valimise definitsioon

Valimise korraldaja defineerib valimise. Eesti riiklikel valimistel jagunevad kõik hääleõiguslikud isikud ühte või mitmesse valimisringkonda. Konkreetsesse ringkonda kuuluval valijal on võimalik hääletamisel valida ainult selle ringkonna kandidaatide vahel.

Valimise defineerimiseks tuleb määratleda vähemalt

1. valimise unikaalne identifikaator ning küsimuste unikaalsed identifikaatorid,
2. täielik loend valimisringkondadest ja -jaoskondadest,
3. hääleõiguslike isikute nimekiri ja jagunemine valimisringkondadesse,
4. kandidaatide nimekiri ja jagunemine valimisringkondadesse.

3.1 Valimise ja küsimuste identifikaatorid

Üht valimist puudutav andmestik on seotud unikaalse valimise identifikaatori abil. Tüüpiliselt hääletatakse ühel konkreetsel valimisel täpselt ühes küsimuses. Siiski eksisteerib võimalus, et valimisel esitatakse mitu küsimust. Kõik küsimused on samamoodi eristatud unikaalse identifikaatori abil.

Identifikaatorite pikkus on piiratud 28 tähemärgiga ASCII kooditabelist. Konkreetse valimise tarbeks kasutatavad identifikaatorid spetsifitseeritakse igakordselt valimise seadistustes. Spetsifikatsiooni realiseerivad rakendused peavad keelduma töötlemast andmeid, mis identifitseerivad valimise/küsimuse, mis ei kuulu rakenduse jaoks seadistatud valimiste/küsimuste nimekirja.

```
election-identifier := 1*28CHAR

question-identifier := 1*28CHAR
```

3.2 Valimisjaoskondade ja -ringkondade nimekiri

Kandidaate on võimalik valimisele üles seada ainult konkreetses valimisringkonnas. Ringkondade abil antakse valijatele hääletamise valikud:

1. Ringkonnad jagunevad jaoskondadeks;
2. Iga valija kuulub talle määratud jaoskonda ja selle kaudu ka ringkonda;
3. Kõigis ühe ringkonna jaoskondades saavad valijad teha valiku vaid selle ringkonna valikute vahel;

Eesti riiklikel valimistel eristatakse Kohalike Omavalitsuste Volikogude (KOV) valimisi, Riigikogu valimisi, Euroopa Parlamendi valimisi ning rahvahääletusi.

KOV korraldatakse valimised vastavalt seadusele „Kohaliku omavalitsuse volikogu valimise seadus“ [KOVVS]. Valimine toimub kohaliku omavalitsuse tasandil, igal omavalitsusel on oma hääletamistulemus. Valimisringkonnad moodustatakse omavalitsuse tasemel vastavalt seaduses kirjeldatud reeglitele.

Riigikogu valimised korraldatakse vastavalt seadusele „Riigikogu valimise seadus“ [RKVS]. Valimine toimub riigi tasandil, hääletamistulemus on kõigile kohalikele omavalitsustele ühine. Riik jaguneb 12 valimisringkonnaks.

Europarlamendi valimised korraldatakse vastavalt seadusele „Euroopa Parlamendi valimise seadus“ [EPVS]. Valimine toimub riigi tasandil, hääletamistulemus on kõigile kohalikele omavalitsustele ühine. Terve riik on üks suur valimisringkond.

Rahvahääletused korraldatakse vastavalt seadusele „Rahvahääletuse seadus“ [RHS]. Valimine toimub riigi tasandil, hääletamistulemus on kõigile kohalikele omavalitsustele ühine. Terve riik on üks suur valimisringkond.

Erinevad valimised ei erine elektroonilise hääletamise andmevormingute ja protseduuride poolest. Erinevad ringkondade jaotused hallatakse Valimiste Infosüsteemi poolt.

Kandidaate on võimalik valimisele üles seada ainult konkreetses valimisringkonnas. Ringkonnad jagunevad jaoskondadeks ning valijad jaotatakse jaoskondade vahel. Kõigis ühe ringkonna jaoskondades saavad konkreetse jaoskonna alla kuuluvad valijad hääletada selle ringkonna kandidaatide poolt. Valija jaoskonnakuuluvuse kaudu on määratud ka tema ringkonnakuuluvus. Valija saab teha valiku ainult tema ringkonnas kandideerivate valikute vahel.

Kuna kohaliku omavalitsuse volikogude valimisel toimub valimine Eesti omavalitsuste (vallad, linnad) tasemel, siis kasutatakse elektroonilise hääletamise protokollistikus valimisringkondade, ja -jaoskondade kirjeldamisel ning valijate ja valikute ringkonnakuuluvuse näitamisel Eesti haldus- ja asustusjaotuse klassifikaatorit EHAK

- Tallinna linna Pirita linnaosa EHAK kood on 0596.
- Aegviidu valla EHAK kood on 0112.

Riigi tasemel toimuvatel valimistel pannakse ringkonna EHAK koodiks kokkuleppeliselt 0. Valimisjaoskondade EHAK koodiks pannakse selle omavalitsuse kood, mille koosseisus konkreetne jaoskond on moodustatud.

Riigikogu ja europarlamendi valimistel ning rahvahääletusel moodustatakse valimisjaoskondade ja –ringkondade nimekirjas igasse ringkonda fiktiivne jaoskond välismaal hääletajate tarbeks. Välismaalaste puhul valimisjaoskonna number on 0 ning vastav EHAK kood on samuti 0.

```
ehak-code = 1*10DIGIT

ehak-district = ehak-code
no-district = 1*10DIGIT

ehak-station = ehak-code
no-station = 1*10 DIGIT

district = ehak-district '.' no-district
district-legacy = ehak-district TAB no-district

station = ehak-station '.' no-station
station-legacy = ehak-station TAB no-station TAB district-legacy
```

Ringkondade nimekirja JSON vorming on defineeritud järgnevalt. Objekti `region_dict` elemente indekseeritakse elemendiga tüüpi `ehak-code`. Objekti `district_dict` elemente indekseeritakse elemendiga tüüpi `district`. Massiivi `stations` elemendid on tüüpi `station`.

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",

  "definitions": {
      "region" : {
          "type": "object",
          "properties": {
              "state": { "type": "string" },
              "county": { "type": "string" },
              "parish": { "type": "string" }
          },
          "additionalProperties": false,
          "minProperties": 1
      },

      "region_dict": {
          "type": "object",
          "patternProperties": {
              "^[0-9]+$": {
                  "$ref": "#/definitions/region"
              }
          },
          "additionalProperties": false,
          "minProperties": 1
      },

      "station": {
```

```
            "type": "string",
            "pattern": "^[0-9]+.[0-9]+$"
        },

        "district": {
            "type": "object",
            "properties": {
                "name": { "type": "string" },
                "stations": {
                    "type": "array",
                    "items": {
                        "$ref": "#/definitions/station"
                    }
                }
            },
            "required": ["stations"]
        },

        "district_dict": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/district"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
        }

    },

     "type": "object",
     "properties": {
         "election": {"type": "string"},
         "districts": {
             "$ref": "#/definitions/district_dict"
         },
         "regions": {
             "$ref": "#/definitions/region_dict"
         }
    },
     "required": ["districts", "regions", "election"],
     "additionalProperties": false
  }
```

Näide:

```
{
  "districts": {
    "164.1": {
```

```
      "name": "Valimisringkond nr. 1",
      "stations": [
        "164.1"
      ]
    },
    "296.1": {
      "name": "Valimisringkond nr. 1",
      "stations": [
        "296.1",
        "296.2"
      ]
    },
    "784.6": {
      "name": "Valimisringkond nr. 6",
      "stations": [
        "524.68"
      ]
    },
    "784.8": {
      "name": "Valimisringkond nr. 8",
      "stations": [
        "614.85",
        "614.86",
        "614.87"
      ]
    },
    "795.1": {
      "name": "Valimisringkond nr. 1",
      "stations": [
        "795.1",
        "795.2"
      ]
    }
  },
  "election": "TESTKOV",
  "regions": {
    "164": {
      "county": "Ida-Viru maakond",
      "parish": "Avinurme vald",
      "state": "Eesti Vabariik"
    },
    "296": {
      "county": "Harju maakond",
      "parish": "Keila linn",
      "state": "Eesti Vabariik"
    },
    "524": {
      "county": "Tallinn",
      "parish": "N\u00f5mme linnaosa",
      "state": "Eesti Vabariik"
```

```
    },
    "614": {
      "county": "Tallinn",
      "parish": "P\u00f5hja-Tallinna linnaosa",
      "state": "Eesti Vabariik"
    },
    "784": {
      "county": "Tallinn"
    },
    "795": {
      "county": "Tartu linn",
      "state": "Eesti Vabariik"
    }
  }
}
```

Ringkondade nimekiri saadakse Valimiste Infosüsteemist ning JSON vormingus faili vahendatakse elektroonilise hääletamise süsteemile BDOC vormingus digitaalallkirjastatud failina.

3.3 Valijate nimekiri

Valijate nimekiri sisaldab valijate nimesid, isikukoode, valimisjaoskonda ning rea numbrit valimisjaoskonna valijate nimekirjas, milles valija hääletab. Valijate nimekiri laaditakse süsteemi digitaalselt allkirjastamata dokumendina, mille vorming on järgmine

```
voter-personalcode = 11DIGIT
voter-name = 1*100UTF-8-CHAR
action = "lisamine" | "kustutamine"
line-no = "" | 1*11DIGIT
reason = "" | "tokend" | "jaoskonna vahetus" | "muu"

voter = voter-personalcode TAB voter-name TAB action TAB station-
 ↪legacy TAB line-no TAB reason LF


version-no = "1"
list-type = "algne" | "muudatused"
voter-list = version-no LF election-identifier LF list-type LF␣
 ↪*voter
```

Pärandsüsteemide andmestruktuurid sisaldavad välja versiooninumber, mille pikkus on piiratud 2 tähemärgiga. Välja väärtus on 1.

Valijate nimekiri võib olla kas algne nimekiri või muudatusnimekiri. Algne nimekiri lubab ainult valijate lisamisi, muudatusnimekirja korral on võimalik ka valijate eemaldamine nimekirjast. Valijakirje võib sisaldada täiendavat informatsiooni – valijakirje reanumbrit

jaoskonna nimekirjas ning põhjust konkreetses muudatusnimekirjas esinemiseks.

Andmete sisu on järgmine.

1. Tüüp “algne” tähistab seda esialgset suurt nimekirja, mis laetakse süsteemi enne e-hääletamise algust ja “muudatused” hilisemaid kumulatiivseid uuendusi.
2. Tegevus “lisamine” tähendab uue valija lisamist nimetatud valimisjaoskonda ja “kustutamine” eemaldamist. Kui valija liigub ühest jaoskonnast teise, siis kantakse valijate nimekirja muudatuste hulka üks kustutamise kirje, millega valija oma eelmisest valimisjaoskonnast kustutatakse ja üks lisamise kirje, millega valija uues valimisjaoskonnas valijate nimekirja kantakse. Algses nimekirjas on kõik kirjed “lisamine” tüüpi.
3. Jaoskond identifitseerib jaoskonna, ringkonna ja omavalitsuse, kus valija hääletab.
4. Rea-number inimese rea number valimisjaoskonna nimekirjas. Täidetud ainult algse nimekirja puhul, muudatuste korral on see väli tühi.
5. Põhjus kasutatakse kustutamiskirjete juures märkimaks kustutamise põhjust. Lisamiskirjete korral peab põhjus tühi olema. Kui põhjuseks on *tokend* tähendab see, et muudatuse rakendumisest alates ei tohi vastava isikukoodiga valija enam hääletada. Kui põhjuseks on *jaoskonna vahetus* tähendab see, et valija kustutatakse ühest jaoskonnast, kuna ta lisatakse teise jaoskonda. Sellisel juhul peab kaasnema kustutamiskirjega ka lisamiskirje (seda kontrollitakse). Kui kasutaja eemaldatakse nimekirjast mingil muul põhjusel (surm, mujale (piirkonda, mis ei osale valimistel) elama kolimine), siis peab põhjuseks olema *muu* või võib põhjus tühi olema. Väli on informatiivne.

Valijate nimekirja signatuur

Valijate nimekiri saadakse SMIT poolt hallatavast Rahvastikuregistrist. Pärandvormingus tekstifailile kaasatakse allkirjafail, mille moodustab Rahvastikuregister võttes algsest valijate nimekirjast SHA256 räsi ning allkirjastades selle räsi 2048 bitise RSA võtmega. Rahvastikuregistri poolt genereeritud avalik võti tehakse kättesaadavaks elektroonilise hääletamise infosüsteemile ning selle võtme alusel kontrollitakse valijate nimekirjade terviklust. Skeem on kasutusel aasta 2015 Riigikogu valimistest.

Rakendatud nimekirja versioon

Rakendatud valijate nimekiri mingis ajahetkes sõltub algnimekirjast ja milliseid muudatusi ning millises järjekorras on rakendatud. Selle seisu ühtlaseks tuvastamiseks tuleb arvutada nimekirja versioon.

NB! See versioon ei ole seotud nimekirja failis sisalduva versiooninumbriga, mis määrab nimekirja formaadi versiooni.

Versiooni arvutamine on järgmine:

```
v_0 = ""
v_n = base64(sha256(v_{n-1} | base64(sha256(nk_n))))
```

kus `nk_n` on `n`-is laetud nimekiri (lugemine algab ühest ehk algnimekiri on `nk_1`), `v_n` on valijate nimekirja versioon pärast selle laadimist, `"` on tühi sõne ja `|` on sõnede sidurdamine operatsioon.

Rakendatud nimekirja versiooni üle peavad arvet kogumisteenus ja töötlemisrakendus, mis garanteerivad, et konkreetne hääl läheks arvesse õiges ringkonnas.

3.4 Valikute nimekiri

Valikute nimekiri sisaldab andmeid kandidaatide (valimistel) või vastusevariantide (rahvahääletusel) kohta. Valimiste korral on lisaks kandidaadi andmetele nimekirjas ka tema valimisnimekirja nimi.

Valijale elektroonilise hääletamise käigus nähtavaid valimiste vahelisi süsteemseid erinevusi on kolm:

1. Rahvahääletusel ei valita erakondadesse kuuluvate kandidaatide vahel vaid vastatakse „JAH“/“EI“ konkreetsetele küsimustele.
2. Riigikogu, KOV ja Euroopa Parlamendi valimistel antakse hääl ühele kandidaadile, kes võib, aga ei pruugi kuuluda suuremasse erakonda/nimekirja.

Protokollistik kodeerib valija võimalikud valikud ringkonnas kuni 11-kohalise arvväärtusena, mis valikute nimekirjas kodeeritakse koos ringkonna EHAK koodiga. Valijale tohivad kättesaadavad olla ainult tema ringkonnakohased valikud. Valijarakendus peab seda omadust tagama ning hääletamistulemust arvutav rakendus kontrollima.

```
choice-no = 1*11DIGIT
district-choice = ehak-district '.' choice-no
```

Valikute nimekirja JSON vorming on defineeritud järgnevalt. Objekti `district_dict` elemente indekseeritakse elemendiga tüüpi `district`. Objekti `list-choices` elemente indekseeritakse elemendiga tüüpi `district-choice`.

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",

    "definitions": {
        "choice": {
            "type": "string"
        },
        "list_choices": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/choice"
```

```
                    }
                },
                "additionalProperties": false,
                "minProperties": 1
            },
            "district_choices" : {
                "type": "object",
                "additionalProperties": {
                    "$ref": "#/definitions/list_choices"
                },
                "minProperties": 1
            },
            "district_dict": {
                "type": "object",
                "patternProperties": {
                    "^[0-9]+.[0-9]+$": {
                        "$ref": "#/definitions/district_choices"
                    }
                },
                "additionalProperties": false,
                "minProperties": 1
            }
        },

        "type": "object",
        "properties": {
            "election": {"type": "string"},
            "choices": {
                "$ref": "#/definitions/district_dict"
            }
        },
        "required": ["election", "choices"],
        "additionalProperties": false
}
```

Näide:

```
{
  "choices": {
    "164.1": {
      "Nimi Valimisliit": {
        "164.126": "Nimi Kandidaat",
        "164.127": "Nimi Kandidaat"
      }
    },
    "296.1": {
      "Nimi Erakond": {
        "296.198": "Nimi Kandidaat",
        "296.199": "Nimi Kandidaat",
        "296.200": "Nimi Kandidaat"
```

```
      },
      "Nimi Valimisliit": {
        "296.115": "Nimi Kandidaat",
        "296.116": "Nimi Kandidaat",
        "296.117": "Nimi Kandidaat"
      },
      "Üksikkandidaadid": {
        "296.101": "Nimi Kandidaat",
        "296.102": "Nimi Kandidaat"
      }
    }
  },
  "election": "TESTKOV"
}
```

PEATÜKK 4

Elektrooniline hääl

IVXV hääletamisprotokoll baseerub topeltümbrikuskeemil, mis tähendab et valija avakujul tahteavaldus krüpteeritakse valimise korraldaja poolt levitatud avaliku võtmega. Krüpteeritud tahteavaldus allkirjastatakse digitaalselt valija käsutuses oleva allkirjastamisvahendiga ning edastatakse kogumisteenusesse mingis kokkulepitud konteinervormingus. Kogumisteenus võib valija poolt allkirjastatud häält täiendavalt kvalifitseerida, veendudes näiteks allkirjastamissertifikaadi kehtivuses. IVXV protokollistik näeb mh. ette kogumisteenuse poolt vastuvõetud häälte registreerimise välises registreerimisteenuses.

Kogumisteenuse poolt talletamisele võetud hääl koos kvalifitseerivate elementidega tehakse kättesaadavaks nii valijarakendusele kui kontrollrakendusele, mis teostavad üksiku hääle peal samad kontrollid, mida hilisem valimise korraldaja töötlemisrakendus teostab kõigi häälte peal. Kvalifitseerivate elementide kontrollimise võimalus annab valijale kindluse, et tema häält on hilisemates protsessides korrektselt menetleda.

4.1 Valija tahteavaldus avakujul

Valija tahteavaldus avakujul eksisteerib valijarakenduses ning hiljem ka kontrollrakenduses. Tahteavaldus sisaldab nii valiku koodi ringkonnas, ringkonna EHAK koodi kui ka valiku nimekirja nime ning konkreetse valiku nime nimekirjas.

```
choice-name = 1*100UTF-8-CHAR
choicelist-name = 1*100UTF-8-CHAR

ballot = district-choice '\x1F' choicelist-name '\x1F' choice-name
```

4.2 Krüpteeritud sedel

Valija tahteavaldus avakujul `ballot` krüpteeritakse valijarakenduse poolt valimise korraldaja genereeritud avaliku võtmega. IVXV vajab krüpteerimiseks mitte-

deterministlikku, homomorfset avaliku võtme krüptosüsteemi. Selliseks süsteemiks sobib ElGamal krüptosüsteem, mida täna rakendatakse IVXV kontekstis jäägiklassi rühmal.

ElGamal avalik võti kodeeritakse koos ElGamal krüptosüsteemi parameetritega ning konkreetset valimist iseloomustava identifikaatoriga. Krüptosüsteemi parameetrid on osaks algoritmi identifikaatori struktuurist, avalik võti on kodeeritud `SubjectPublicKeyInfo` struktuuri.

```
elGamalEncryption OBJECT IDENTIFIER ::= {
    {iso(1) org(3) dod(6) internet(1) private(4) enterprise(1)␣
↪dds(3029) asymmetric-encryption(2) 1}
}

elGamal-Params-IVXV ::= SEQUENCE {
    p                   INTEGER,
    g                   INTEGER,
    election-identifier GeneralString
}

elGamalPublicKey ::= SEQUENCE {
    y           INTEGER,
}

SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm   AlgorithmIdentifier,
    subjectPublicKey    BIT STRING
}
```

Valija tahteavalduse krüpteerimiseks võetakse UTF-8 kodeeringus struktuur `ballot` ning teisendatakse see ElGamal parameetrite poolt kirjeldatud rühma elemendiks. Eeldame, et parameeter `p` on 256 baiti. Sellisel juhul võib struktuuri `ballot` pikkus olla 253 baiti. Avakujul tahteavaldus pikendatakse parameetri `p` pikkuseni.

```
padded-ballot = ballot '\x00' '\x01' *'\xff' '\x00'
```

Pikendatud tahteavaldust interpreteeritakse kui täisarvu, mis kodeeritakse ruutjäägina parameetri `p` poolt kirjeldatud rühmas. Kodeerimine on üksühene ning oluline krüptogrammi edasise miksimise jaoks.

Tahteavaldus krüpteeritakse vastavalt ElGamal meetodile avaliku võtmega.

```
elGamalEncryptedMessage ::= SEQUENCE {
    a           INTEGER,
    b           INTEGER
}

encryptedBallot ::= SEQUENCE {
    algorithm   AlgorithmIdentifier,
    cipher  ANY
}
```

Andmestruktuuri `encryptedBallot` DER-kodeering on krüpteeritud sedel ehk sisemine ümbrik topeltümbriku skeemis.

Tahteavalduse krüpteerimise käigus genereeritakse valijarakenduses juhuarv, mida ElGamal krüpteerimisel kasutab. Sama juhuarv avalikustatakse hiljem kontrollrakendusele. Tulenevalt ElGamal krüptosüsteemi eripärast funktsioneerib see juhuarv nö. teise võtmena ning võimaldab krüptogrammi dekodeerimist kontrollrakenduses.

4.3 Valija poolt allkirjastatud hääl

Krüpteeritud sedel tuleb enne kogumisteenusesse talletamisele saatmist digitaalselt allkirjastada, milleks on võimalik kasutada kõiki Eesti Vabariigis kehtivaid digitaalallkirjavahendeid – ID-kaart, Digi-ID, Mobiil-ID.

Käesolev spetsifikatsioon näeb ette Eesti Vabariigi Standardikavandis [BDOC2.1] defineeritud BDOC allkirjavormingu kasutamise. BDOC allkirjavorming koosneb ETSI standardi TS 101 903 (XadES) profiilist ning OpenDocument konteineri vormingust. IVXV protokollistik võimaldab ka alternatiivsete allkirja- ning konteinervormingute kasutamist.

Olenevalt käimasoleval valimisel esitatud küsimuste arvust võib digitaalselt allkirjastatud hääl sisaldada ühte või mitut andmefaili MIME tüübiga application/octet-stream. Iga andmefaili sisuks on krüpteeritud sedel. Andmefaili ja teiste signeeritavate andmeobjektide räsimiseks enne allkirjastamist kasutatakse räsifunktsiooni SHA-256. Andmefaili nimi moodustatakse laiendist ‘ballot’ ning valimise ja küsimuse identifikaatorist. Kõik viidatud andmefailid peavad allkirjakonteineris sisalduma. Digitaalselt allkirjastatud hääl ei tohi sisaldada muid andmefaile kui neid, mis sisaldavad hääli mõne käimasoleva valimise kontekstis. Seadistusele mittevastavate häälte vastuvõtmisest, talletamisest ja töötlemisest peab kogumisteenus keelduma.

```
extension = "ballot"

encrypted-ballot-name = election-identifier '.' question-identifier
↪'.' extension
```

Valija poolt valijarakenduses allkirjastatud hääl moodustatakse selliselt, et on võimalik selle edasine kvalifitseerimine kogumisteenuses. Käesolev septsifikatsioon näeb ette hääle kvalifitseerimiseks nii OCSP kehtivuskinnituse kui PKIX ajatempli võtmise. Sellisena on lõplik, kvalfitseeritud hääl, BDOC-TS vormingus.

Kui hääl allkirjastatakse ID-kaardi või Digi-ID’ga, siis toimub algse allkirjastatud konteineri moodustamine valijarakenduses. Kui hääl allkirjastatakse Mobiil-ID’ga, siis toimub konteineri moodustamine valijarakenduse ning kogumisteenuse poolt vahendatava DigiDoc-teenuse koostöös. Mobiil-ID juhtumil kasutab kogumisteenus DigiDoc-teenust ainult signatuuri saamiseks krüpteeritud sedelile. Kõik hääle kvalifitseerimiseks vajalikud elemendid hangitakse vastavatelt teenustelt alles siis kui valijarakendus on saatnud signeeritud hääle talletamiseks. Kvalifitseeritud hääl esitatakse kogumisteenuse poolt valijarakendusele verifitseerimiseks, ainult kvalifitseeritud hääl peab vastama

BDOC 2.1 standardi tingimustele – valijarakenduse poolt moodustatud hääl on vaheetapp kvalifitseeritud hääleni jõudmiseks.

Valijarakenduses signeeritud häälel peab olema üks ja ainult üks allkiri, mida hoitakse signatuurifailis META-INF/signature0.xml. Häält ja allkirja sisaldav konteiner moodustatakse BDOC 2.1 standardis kirjeldatud meetodit kasutades.

Täpsustame valijarakenduses allkirjastatud hääle ühe küsimuse korral.

Räsialgoritmina *DIGEST_ALG* on kasutusel SHA-256 (http://www.w3.org/2001/04/xmlenc#sha256). XML kanoniseerimiseks (*CANON_ALG*) kasutatakse meetodit c14n11 (http://www.w3.org/2006/12/xml-c14n11).

RSA võtmete korral (ID-kaart, Digi-ID) on allkirjastamismeetodiks http://www.w3.org/2001/04/xmldsig-more#rsa-sha256. ECC võtmete korral (Mobiil-ID) http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256.

Identifikaatorite *VOTE_REF*, *SP_URI* ning *SV_URI* täpne väärtus ei ole oluline.

Element *SignedProperties*

Element *SignedProperties* moodustatakse kooskõlas BDOC 2.1 standardiga. Kui kvalifitseerimisel kasutatakse ajatemplit, siis elementi *SignaturePolicyIdentifier* ei kasutata. Ühtegi mitte-kohustuslikku elementi ei kasutata. Allkirjastamise kellaaja fikseerib andmestruktuuri täitev arvuti ning valija X509 sertifikaat saadakse kas ID-kaardilt või DigiDoc-teenuse vahendusel.

```
<xades:SignedProperties xmlns:asic="http://uri.etsi.org/02918/v1.2.1
↪#" xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xades=
↪"http://uri.etsi.org/01903/v1.3.2#" Id="%SP_URI%">
<xades:SignedSignatureProperties>
  <xades:SigningTime>%SIGNING_TIME%</xades:SigningTime>
  <xades:SigningCertificate>
    <xades:Cert>
      <xades:CertDigest>
        <ds:DigestMethod Algorithm="%DIGEST_ALG%"></ds:DigestMethod>
        <ds:DigestValue>%CERT_DIGEST%</ds:DigestValue>
      </xades:CertDigest>
      <xades:IssuerSerial>
        <ds:X509IssuerName>%ISSUER_NAME%</ds:X509IssuerName>
        <ds:X509SerialNumber>%ISSUER_SERIAL%</ds:X509SerialNumber>
      </xades:IssuerSerial>
    </xades:Cert>
  </xades:SigningCertificate>
</xades:SignedSignatureProperties>
<xades:SignedDataObjectProperties>
  <xades:DataObjectFormat ObjectReference="#%VOTE_REF%">
    <xades:MimeType>application/octet-stream</xades:MimeType>
  </xades:DataObjectFormat>
</xades:SignedDataObjectProperties>
</xades:SignedProperties>
```

Element *SignedInfo*

Element *SignedInfo* moodustatakse kooskõlas BDOC 2.1 standardiga viidates nii krüpteeritud sedelile (*VOTE_DIGEST*) kui elemendile *SignedProperties* (*SP_DIGEST*).

```
<ds:SignedInfo xmlns:asic="http://uri.etsi.org/02918/v1.2.1#" xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xades="http://uri.etsi.org/01903/v1.3.2#">
  <ds:CanonicalizationMethod Algorithm="%CANON_ALG%"></ds:CanonicalizationMethod>
  <ds:SignatureMethod Algorithm="%SIG_ALG%"></ds:SignatureMethod>
  <ds:Reference Id="%VOTE_REF%" Type="" URI="%VOTE_URI%">
    <ds:DigestMethod Algorithm="%DIGEST_ALG%"></ds:DigestMethod>
    <ds:DigestValue>%VOTE_DIGEST%</ds:DigestValue>
  </ds:Reference>
  <ds:Reference Type="http://uri.etsi.org/01903#SignedProperties" URI="#%SP_URI%">
    <ds:Transforms>
      <ds:Transform Algorithm="%CANON_ALG%"></ds:Transform>
    </ds:Transforms>
    <ds:DigestMethod Algorithm="%DIGEST_ALG%"></ds:DigestMethod>
    <ds:DigestValue>%SP_DIGEST%</ds:DigestValue>
  </ds:Reference>
</ds:SignedInfo>
```

Element *SignatureValue*

Element *SignatureValue* moodustatakse kooskõlas BDOC 2.1 standardiga. Kanoniseeritud elemendist *SignedInfo* arvutatakse räsi, mis allkirjastatakse PKCS1 meetodiga.

```
<ds:SignatureValue xmlns:asic="http://uri.etsi.org/02918/v1.2.1#" xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Id="%SV_URI%">%SIG_VALUE%</ds:SignatureValue>
```

Element *XAdESSignatures*

Element *XAdESSignatures* sisaldab ühte *Signature* elementi, mis on koostatud lähtudes kõigist eelmistest elementidest ning valija X509 sertifikaadist. Elementi *UnsignedProperties* ei kasutata.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<asic:XAdESSignatures xmlns:asic="http://uri.etsi.org/02918/v1.2.1#">
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="S0">
```

```
        %SI_XML%
        %SV_XML%
    <ds:KeyInfo>
      <ds:X509Data>
          <ds:X509Certificate>%X509_CERT%</ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
    <ds:Object>
        <xades:QualifyingProperties xmlns:xades="http://uri.etsi.
↪org/01903/v1.3.2#" Target="#S0">
            %SP_XML%
      </xades:QualifyingProperties>
    </ds:Object>
  </ds:Signature>
</asic:XAdESSignatures>
```

PEATÜKK 5

Elektroonilise hääle kvalifitseerimine talletamiseks

5.1 Kvalifitseeritud hääl

Valijarakenduse töö tulemusena saadetakse kogumisteenusesse talletamiseks topeltümbrik, mis sisaldab endas valija tahteavaldust krüpteeritud kujul, valija allkirja krüpteeritud tahteavaldusel kooskõlastatud allkirja- ja konteinervormingus ning valija allkirjastamissertifikaati X509 vormingus.

Hääle edukaks talletamiseks näeb IVXV protokoll ette hääle registreerimise välise registreerimisteenuse osutaja juures ning registreerimistõendi valijarakendusele kättesaadavaks tegemise. Valimise korraldaja võib hääle kvalifitseerimiseks näha ette täiendavaid samme lisaks registreerimisele – näiteks kehtivuskinnituse hankimist hääle allkirjastanud sertifikaadi kohta.

Kõik kogumisteenuse poolt hangitavad kvalifitseerivad elemendid, mis määravad hääle staatuse hilisemates töötlusetappides tuleb esitada valijarakendusele ning nõudmise korral ka kontrollrakendusele tagamaks, et valija saab oma hääle korrektse menetlemise võimalikkusest õigeaegselt teada.

OCSP kehtivuskinnitus

OCSP (Online Certificate Status Protocol) on standartne protokoll X509 sertifikaatide kehtivusinfo pärimiseks. Kogumisteenus võib seda protokolli kasutada hääle allkirjastanud sertifikaadi kehtivuse teada saamiseks. OCSP vastus ütleb, et sertifikaat kehtis päringu tegemise ajahetkel, kuid ei seosta OCSP vastust konkreetse allkirjaga.

OCSP-TM kehtivuskinnitus

BDOC 2.1 standard kirjeldab BDOC-TM profiili, kus OCSP protokolliga hangitud kehtivuskinnitus toimib ka ajamärgendina, mis kinnitab, et konkreetne allkiri eksisteeris enne OCSP kehtivuskinnituse võtmist.

RFC3161 ajatempel

RFC3161 ajatempli protokolliga saadakse usaldusteenuse pakkujalt kinnitus, et mingi andmekogum eksisteeris enne teatud ajahetke. BDOC-TS kontekstis ajatembeldatakse allkirja element *SignatureValue* kanoniseeritud kujul. Klassikaline OCSP vastus koos RFC 3161 vormingus ajatempliga kvalifitseerivad BDOC-TS allkirja.

5.2 Hääle registreerimine

IVXV registreerimisprotokoll on kirjeldatud dokumendis “Elektroonilise hääletamise infosüsteemi IVXV registreerimisteenus”. Registreerimisteenus toimub RFC3161 ajatempli protokolli baasil. Protokolli on laiendatud selliselt, et kogumisteenus saab ajatempli päringule anda oma signatuuri, mis teeb võimalikuks hilisema võrdleva väljavõtte registreerimisteenusest. Sõltumatu registreerimisteenuse olemasolu vähendab häälte kogumisteenuse poolt ‘kaotamise’ riski.

Puudub olemuslik vajadus registreerimisprotokolli sidumiseks ajatempli protokolliga.

5.3 Talletamine

Elektroonilise hääle talletamine kogumisteenuses tähendab

1. hääle vastuvõtmist valijarakenduselt ning hääletaja allkirja verifitseerimist;
2. hääle võimalikku kvalifitseerimist – näiteks sertifikaadi kehtivuse tõendamist hääle allkirjastamisele lähedasel ajahetkel;
3. hääle registreerimist sõltumatus registreerimisteenuses;
4. häält kvalifitseerivate elementide vahendamist valijarakendusele.

Erinevad kombinatsioonid allkirjavormingust ning hääli kvalifitseerivatest teenustest võivad tekitada erinevaid IVXV-profiile. Konkreetse dokumendi raames on IVXV profiil:

1. Allkirjastatud hääle vorming on BDOC-TS.
2. Kehtivuskinnitusprotokolliks on standartne OCSP.
3. BDOC-TS kvalifitseerimiseks kasutatav RFC3161 ajatempel on kasutusel ka registreerimistõendina.

PEATÜKK 6

Elektroonilise hääle kontrollimine

Elektroonilist häält kontrollitakse töötlemisrakenduses, kogumisteenuses, valijarakenduses ja kontrollrakenduses. Kõige põhjalikuma kontrolli läbib elektrooniline hääl e-urni koosseisus töötlemisrakenduses, kus otsustatakse konkreetse hääle lugemisele saatmine või mittesaatmine. Iga üksiku hääle kohta läbitakse töötlemisrakendusega analoogsel tasemel kontroll valijarakenduses, kus veendutakse, et kogumisteenus on hääle kvalifitseerinud selliselt, et töötlemisrakenduses tehtavad kontrollid õnnestuvad. Valijarakendusega analoogsed kontrollid viib läbi kontrollrakendus.

6.1 Kontrollid kogumisteenuses

Valijarakendus saadab kogumisteenusele allkirjastatud hääle koosseisus

1. krüpteeritud sedeli
2. valija allkirja krüpteeritud sedelil
3. valija allkirjastamissertifikaadi

Kogumisteenus viib läbi minimaalselt järgmised kontrollid:

1. hääle allkirjastaja on valijate nimekirjas,
2. allkirjastatud hääl on esitatud korrektses konteinervormingus,
3. digitaalallkiri krüpteeritud sedelil on korrektne,
4. hääle allkirjastaja sertifikaat on kehtiv hääle vastuvõtmise ajahetkel.

Hääle allkirjastaja sertifikaadi kehtivuse kontrolliks teeb kogumisteenus päringu kehtivuskinnitusteenusele. Kogumisteenus verifitseerib kehtivuskinnitusteenuse vastust sertifikaadi oleku kohta ning lisab selle vastuse häält kvalifitseerivate elementide hulka.

Kogumisteenus registreerib hääle talletamise fakti välises registreerimisteenuses allkirjastades registreerimispäringu ning talletades registreerimisteenuse poolt allkirjastatud registreerimistõendi häält kvalifitseerivate elementide hulka.

Kogumisteenus tagastab kõik tema poolt hangitud häält kvalifitseerivad elemendid valijarakendusele koos hääle unikaalse identifikaatoriga.

6.2 Kontrollid valijarakenduses

Valijarakendus moodustab valija avakujul tahteavalduse põhjal krüpteeritud sedeli ning allkirjastab selle valija allkirja andmise vahendiga.

Valijarakenduse rolliks peale hääle allkirjastamist on veenduda, et kogumisteenus käitus häält kvaliftseerivate elementide võtmisel protokollikohaselt ning et hääl on talletatud selliselt, et ta saab töötlemisrakenduse poolt arvesse võetud.

Valijarakendus viib läbi minimaalselt järgmised kontrollid:

1. Kogumisteenus võttis kehtivuskinnituse valija sertifikaadile volitatud kehtivuskinnitusteenuselt. Valijarakendus kontrollib allkirja kehtivuskinnitusteenuse vastusel.
2. Kogumisteenus registreeris valija poolt allkirjastatud hääle volitatud registreerimisteenuses. Valijarakendus kontrollib, et kogumisteenuse poolt moodustatud päring oli kogumisteenuse poolt signeeritud ning viitas korrektselt allkirjastatud häälele. Valijarakendus kontrollib, et registreerimisteenuse vastus on allkirjastatud õige registreerimisteenuse osutaja poolt ning selles sisaldub kogumisteenuse poolt allkirjastatud päring.

Kui hääle kvalifitseerimiseks vajalike elementide kontroll ei õnnestu, siis teavitab valijarakendus sellest kasutajat.

6.3 Kontrollid kontrollrakenduses

Kontrollrakendus saab valijarakendusest järgmise info:

1. Krüpteeritud sedeli moodustamisel kasutatud juhuslikkuse
2. Allkirjastatud hääle unikaalse identifikaatori kogumisteenuses

Kontrollrakendus kasutab hääle unikaalset identifikaatorit kogumisteenusest järgmise info saamiseks:

1. krüpteeritud sedel,
2. valija allkiri krüpteeritud sedelil,
3. valija allkirjastamissertifikaat,
4. häält kvalifitseerivad elemendid, kaasaarvatud kehtivuskinnitus ja registreerimistõend

Kontrollrakendus teostab järgmised kontrollid:

1. allkirjastatud hääl on esitatud korrektses konteinervormingus,
2. digitaalallkiri krüpteeritud sedelil on korrektne,

3. hääle allkirjastaja sertifikaat on kehtiv hääle vastuvõtmise ajahetkel, mida kinnitab korrektne kehtivuskinnitus,
4. hääl on korrektselt registreeritud õiges registreerimisteenuses.

Nende kontrollide teostamise järel kuvab kontrollrakendus hääle allkirjastanud isiku andmeid.

Täiendavalt kasutab kontrollrakendus krüpteeritud sedeli moodustamisel kasutatud juhuslikkust krüpteeritud sedeli dekrüpteerimiseks. NB! Ühe hääle krüpteerimisel kasutatud juhuslikkust saab kasutada ainult selle hääle dekrüpteerimiseks. Mitme erineva hääle dekrüpteerimiseks läheb vaja häälte salastamise võtme privaatkomponenti.

Kontrollrakendus veendub, et dekrüpteerimisel saadud avatekst vastab avakujul tahteavalduse vorminõuetele.

Kontrollrakendus kuvab vorminõuetele vastava tahteavalduse võimaldamaks kontrollijal selle tahteavalduse korrektsuses veenduda.

6.4 Kontrollid töötlemisrakenduses

Töötlemisrakendus kontrollib iga üksikut häält eraldi, veendudes muuhulgas, et iga kogumisteenuse ning registreerimisteenuse poolt esitatud vaated e-urni sisu kohta on konsistentsed. Seejärel otsustab töötlemisrakendus iga valija häälte kohta, milline neist on ajaliselt viimane ning suunatakse töötlemise järgmisesse etappi, mille tulemusena hääl võib jõuda lugemisele.

Töötlemisrakenduse sisendiks on:

1. Loend registreerimisteenuse poolt vastuvõetud registreerimispäringutest
2. Loend kogumisteenuses rakendatud valijanimekirjadest
3. Kogumisteenuse poolt üle antud e-urn, mis sisaldab iga hääle kohta krüpteeritud sedelit, valija allkirja krüpteeritud sedelil, valija allkirjastamissertifikaati, sertifikaadi kehtivuskinnitust ning registreerimistõendit.

Töötlemisrakendus kontrollib registreerimisteenuse ja kogumisteenuse kooskõla ning väljastab erinevused:

1. Hääled, mille kohta on olemas registreerimispäring kogumisteenuses, kuid vastus ei ole jõudnud kogumisteenusesse
2. Hääled, mille kohta on olemas registreerimispäring registreerimisteenuses, kuid mida kogumisteenus ei ole üle andnud

Töötlemisrakendus kontrollib iga üksikut häält:

1. hääle allkirjastaja oli valijate nimekirjas,
2. allkirjastatud hääl on esitatud korrektses konteinervormingus,
3. digitaalallkiri krüpteeritud sedelil on korrektne,
4. hääle allkirjastaja sertifikaat on kehtiv hääle vastuvõtmise ajahetkel, mida kinnitab korrektne kehtivuskinnitus,

5. hääl on korrektselt registreeritud õiges registreerimisteenuses.

Töötlemisrakendus otsustab, milline valija häältest oli viimane ning liigub töötlemise järgmisesse etappi. S.t. üks häält kvalifitseerivatest elementidest täidab hääle talletamise aja fikseerimise rolli ning selle elemendi põhjal moodustatakse üksikute häälte ajaline järgnevus. Olenevalt IVXV profiilist võib see element olla kehtivuskinnituse koosseisus (BDOC-TM), eraldi ajatemplina (BDOC-TS) või registreerimistõendi koosseisus (BDOC-TS).

PEATÜKK 7

Suhtlusprotokollid

7.1 Liides

Kogumisteenuse valijale suunatud mikroteenused suhtlevad valijarakendusega ja kontrollrakendusega JSON-RPC protokolli vahendusel.

id JSON-RPC päringuidentifikaator
method RPC meetod
params Konkreetse RPC meetodi parameetrid

```
{
    "id": 0.0,
    "method": "RPC.Method",
    "params": [
        {
            "MethodParam": "value",
            "SessionID": "ec3a0cab353d552952289f2c7ad52e27"
        }
    ]
}
```

error Võimalik veainfo või `null` vea puudumisel
id JSON-RPC päringuidentifikaator, peab ühtima päringus kasutatud id'ga
result Meetodipõhine vastusandmestruktuur

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "ResultParam": "value",
        "SessionID": "ec3a0cab353d552952289f2c7ad52e27"
    }
}
```

Esimese päringuvahetuse käigus mõne IVXV mikroteenusega väljastatakse suhtlevale rakendusele HEX-kodeeritud unikaalne seansiidentifkaator (`result.SessionID`), mida rakendus kasutab edaspidi kõigis kogumisteenuse suunalistes päringutes (`params.SessionID`). Seansiidentifikaatori abil seostatakse erinevad hääletamisega seotud RPC päringud üheks seansiks. Seostamine on informatiivne ning selle eesmärk on logianalüüsi lihtsustamine, hääle ringkonnakuuluvust jm. sisulisi aspekte puudutavad otsused tehakse digiallkirjastatud andmete põhjal.

Transpordiprotokollina on kasutusel TLS. Krüpteeritud kanali termineerimine toimub konkreetses mikroteenuses. Võimaldamaks koormuse jaotamist ning mikroteenuste paindlikku evitamist kasutatakse TLS'i SNI laiendust, mis lubab vahendusteenusel TLS voogu termineerimata õigesse mikroteenusinstantsi suunata. Vahendusteenus on tüüpiliselt kättesaadav kogumisteenuse välise liidese 443 pordis.

7.2 Valikute nimekirja hankimine

Valikute nimekirja hankimine tähendab valijarakenduse suhtlemist nimekirjateenusega (SNI choices.ivxv.invalid). Valikute nimekirja hankimine eeldab valija autentimist ning tema ringkonnakuuluvuse tuvastamist.

Valijarakendus teeb päringu `RPC.VoterChoices` nimekirjade hankimiseks.

params.AuthMethod Toetatud valikud on meetodid `tls` ja `ticket`.

params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.

Päring `RPC.VoterChoices` ID-kaardiga autentimise korral - autentimine toimub TLS protokolli tasemel päringu töötlemise ajal kasutades ID-kaardi autentimissertifikaati.

```
{
    "id": 0.0,
    "method": "RPC.VoterChoices",
    "params": [
        {
            "AuthMethod": "tls",
            "OS": "Operating System,2,0"
        }
    ]
}
```

Päring `RPC.VoterChoices` Mobiil-ID'ga autentimise korral - päringu sooritamiseks tuleb eelnevalt kasutada DigiDocService vahendusteenuse (SNI dds.ivxv.invalid) abi allkirjastatud autentimistõendi saamiseks.

params.AuthToken Autentimisteenuse vahendusel allkirjastatud tõend, mis sisaldab endas valija unikaalset identifikaatorit.

params.SessionID Kuna Mobiil-ID korral on nimekirja hankimisele eelnenud interaktsioon autentimistõendi saamiseks, on olemas seansiidentifikaator, mida tuleb kasutada.

```
{
    "id": 0.0,
    "method": "RPC.VoterChoices",
    "params": [
        {
            "AuthMethod": "ticket",
            "AuthToken":
↪"G1RTZqBSBKrzqReuKYrmFUFXWFPvaxhJjdiZi6zqAnaK3OvrT...",
            "OS": "Operating System,2,0",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
        }
    ]
}
```

Nimekirjateenuse vastus päringule `RPC.VoterChoices`.

result.Choices Valija ringkonnakuuluvuse identifikaator `VoterDistrict`

result.List BASE64-kodeeritud ringkonna valikute nimekiri `DistrictChoices`

result.Voted Kui valija on juba hääletanud, siis `true`, vastasel juhul seda välja vastuses ei ole.

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "Choices": "0140.1",
        "List":
↪"ew0KICAgICAgICAgICAgIkVyYWtvbmQgMSI6IHsNCiAgICAgICAgICAgIC...",
        "SessionID": "ec3a0cab353d552952289f2c7ad52e27",
        "Voted": true
    }
}
```

Võimalikud veateated päringu `RPC.VoterChoices` korral.

BAD_CERTIFICATE Viga valija isikutuvastussertifikaadiga.

BAD_REQUEST Vigane päring.

INELIGIBLE_VOTER Valijal pole õigust hääletada.

INTERNAL_SERVER_ERROR Viga serveri sisemises töös.

UNAUTHENTICATED Autentimata päring.

VOTER_TOO_YOUNG Valija on liiga noor.

VOTING_END Hääletusperiood on lõppenud.

7.3 Allkirjastatud hääle saatmine talletamiseks

Allkirjastatud hääle saatmine talletamiseks tähendab valijarakenduse suhtlemist hääletamisteenusega (SNI voting.ivxv.invalid).

Valijarakendus teeb päringu `RPC.Vote` allkirjastatud hääle talletamiseks saatmiseks.

params.AuthMethod Toetatud valikud on meetodid `tls` ja `ticket`.

params.Choices Valija ringkonnakuuluvuse identifikaator `VoterDistrict` mis kehtis valikute nimekirja hankimise ajal. Parameetri korrektne kasutamine võimaldab kogumisteenusel valijat hoiatada kui tema ringkonnakuuluvus on võrreldes hääletamise algushetkega muutunud.

params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.

params.Type Allkirjastatud hääle vorming. Hetkel on ainus toetatud väärtus `bdoc`.

params.Vote BASE64-kodeeritud hääl `SignedVote` eelpoolmääratud vormingus.

Päring `RPC.Vote` ID-kaardiga autentimise korral.

```
{
    "id": 0.0,
    "method": "RPC.Vote",
    "params": [
        {
            "AuthMethod": "tls",
            "Choices": "0140.1",
            "SessionID": "ec3a0cab353d552952289f2c7ad52e27",
            "OS": "Operating System,2,0",
            "Type": "bdoc",
            "Vote":
↪"UEsDBAoABgAAAAIAAAAbWltZXR5cGVhcHBsaWNhdGlv\nbi92bmQuZX..."
        }
    ]
}
```

Päring `RPC.Vote` Mobiil-ID'ga autentimise korral.

```
{
    "id": 0.0,
    "method": "RPC.Vote",
    "params": [
        {
            "AuthMethod": "ticket",
            "AuthToken":
↪"G1RTZqBSBKrzqReuKYrmFUFXWFPvaxhJjdiZi6zqAnaK3OvrT...",
            "Choices": "0919.1",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b",
            "OS": "Operating System,2,0",
            "Type": "bdoc",
            "Vote":
↪"UEsDBAoAAAAAAAAAACKIflFHwAAAB8AAAAIAAAAbWltZXR5cGVhcHB..."
        }
    ]
}
```

Hääletamisteenuse vastus päringule `RPC.Vote`.

result.Qualification.ocsp

result.Qualification.tspreg Kogumisteenuse poolt hangitud täiendavad tõendid valijarakenduse poolt loodud hääle `SignedVote` kvalifitseerimiseks ning korrektseks registreerimiseks. Vastuse koosseis sõltub kogumisteenuse konkreetsest seadistusest, antud juhul kasutatakse standardset OCSP protokolli valija allkirjasertifikaadi kehtivuse kontrolliks ning PKIX ajatempliprotokolli põhist registreerimisteenust nii hääle andmise aja fikseerimiseks kui elektroonilise hääle registreerimiseks välises sõltumatus teenuses. Valijarakendusele kontrollimiseks edastatakse nii OCSP vastus kui PKIX vormingus ajatempel koos registreerimisteenusele vajalike täiendustega.

result.TestVote Kui hääl esitati enne hääletamise algust ning läks arvesse proovihäälena, siis `true`, vastasel juhul seda välja vastuses ei ole. Valijarakendus kuvab valijale proovihääle korral sellekohase hoiatuse.

result.VoteID Hääle identifikaator talletusteenuses, mille alusel on kontrollrakendusel võimalik häält hilisemaks analüüsiks välja nõuda.


```
{
    "error": null,
    "id": 0.0,
    "result": {
        "Qualification": {
            "ocsp":
↪"MIIFTAoBAKCCBUUwggVBBgkrBgEFBQcwAQEEggUyMIIFLjCB5qFMME...",
            "tspreg":
↪"MIIDsAYJKoZIhvcNAQcCoIIDoTCCA50CAQMxCzAJBgUrDgMCGgQS..."
        },
        "SessionID": "ec3a0cab353d552952289f2c7ad52e27",
        "TestVote": true,
        "VoteID": "VM/cUIU4n7VjxpUx1fC00Q=="
    }
}
```


Võimalikud veateated päringu `RPC.Vote` korral.

BAD_CERTIFICATE Viga valija isikutuvastus- või allkirjastamissertifikaadiga.

BAD_REQUEST Vigane päring.

IDENTITY_MISMATCH Isikutuvastus- ning allkirjastamissertifikaadi isikukoodid ei kattu.

INELIGIBLE_VOTER Valijal pole õigust hääletada.

INTERNAL_SERVER_ERROR Viga serveri sisemises töös.

OUTDATED_CHOICES Valija ringkonnakuuluvus on nimekirja hankimisest muutunud.

UNAUTHENTICATED Autentimata päring.

VOTER_TOO_YOUNG Valija on liiga noor.

VOTING_END Hääletusperiood on lõppenud.

7.4 Hääletamine Mobiil-ID'ga

Mobiil-ID kasutamine allkirjastamis- ning autentimisvahendina tingib teenusega DigiDocService liidestuva abiteenuse (SNI dds.ivxv.invalid) kasutamise autentimistõendi hankimiseks enne valikute nimekirja hankimist ning hääle allkirjastamiseks enne talletamist.

Autentimistõendi hankimine

Valijarakendus teeb päringu `RPC.Authenticate` Mobiil-ID autentimise algatamiseks.

params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.
params.PhoneNo Mobiil-ID kasutaja telefoninumber.

```
{
    "id": 0.0,
    "method": "RPC.Authenticate",
    "params": [
        {
            "OS": "Operating System,2,0",
            "PhoneNo": "+37200000766"
        }
    ]
}
```

result.ChallengeID Mobiil-ID kontrollkood kuvamiseks valijarakenduses
result.SessionCode Mobiil-ID seansiidentifikaator edasiste poll-päringute jaoks,

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "ChallengeID": "4004",
        "SessionCode": "2127729011",
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
    }
}
```

Võimalikud veateated päringu `RPC.Authenticate` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
MID_BAD_CERTIFICATE Viga valija Mobiil-ID isikutuvastussertifikaadiga.
MID_NOT_USER Telefoninumber ei kuulu Mobiil-ID kliendile.
VOTING_END Hääletusperiood on lõppenud.

Valijarakendus teeb päringu `RPC.AuthenticateStatus` autentimisprotsessi oleku hindamiseks.

params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.

params.SessionCode Autentimisseansi identifikaator

```
{
    "id": 0.0,
    "method": "RPC.AuthenticateStatus",
    "params": [
        {
            "OS": "Operating System,2,0",
            "SessionCode": "2127729011",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
        }
    ]
}
```

result.AuthToken Autentimistõend teistele IVXV teenustele esitamiseks või `null`, kui päringu töötlemine alles käib.

result.GivenName Eduka autentimise korral valija eesnimi

result.PersonalCode Eduka autentimise korral valija isikukood

result.Status Päringu staatus - POLL viitab vajadusele päringut korrata, OK viitab edukale autentimisele. Vastuse muud väljad sisaldavad infot vaid siis kui väärtus on OK.

result.Surname Eduka autentimise korral valija perekonnanimi

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "AuthToken": null,
        "GivenName": "",
        "PersonalCode": "",
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b",
        "Status": "POLL",
        "Surname": ""
    }
}
```

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "AuthToken":
↪"G1RTZqBSBKrzqReuKYrmFUFXWFPvaxhJjdiZi6zqAnaK3OvrT2Qu6...",
        "GivenName": "MARY \u00c4NN",
        "PersonalCode": "11412090004",
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b",
```

```
        "Status": "OK",
        "Surname": "O\u2019CONNE\u017d-\u0160USLIK"
    }
}
```

Võimalikud veateated päringu `RPC.AuthenticateStatus` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
MID_ABSENT Valija mobiiltelefon ei ole kättesaadav.
MID_CANCELED Valija katkestas Mobiil-ID seansi.
MID_EXPIRED Mobiil-ID seanss on aegunud.
MID_GENERAL Viga Mobiil-ID teenuse töös.
VOTING_END Hääletusperiood on lõppenud.

Hääle allkirjastamine

Valijarakendus teeb päringu `RPC.GetCertificate` allkirjastamissertifikaadi hankimiseks.

params.AuthMethod Toetatud ainult autentimismeetod `ticket`.
params.AuthToken Mobiil-ID autentimistõend.
params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.
params.PhoneNo Hääle allkirjastaja telefoninumber

```
{
    "id": 0.0,
    "method": "RPC.GetCertificate",
    "params": [
        {
            "AuthMethod": "ticket",
            "AuthToken":
↪"G1RTZqBSBKrzqReuKYrmFUFXWFPvaxhJjdiZi6zqAnaK3OvrT...",
            "OS": "Operating System,2,0",
            "PhoneNo": "+37200000766",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
        }
    ]
}
```

result.Certificate Allkirjastamissertifikaat X509 vormingus

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "Certificate":
↪"MIIEVjCCAz6gAwIBAgIQRfmbsIcpkQ9UhxScCwG6VDANBgkqhki...",
```

```
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
    }
}
```

Võimalikud veateated päringu `RPC.GetCertificate` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
MID_BAD_CERTIFICATE Viga valija Mobiil-ID allkirjastamissertifikaadiga.
MID_GENERAL Viga Mobiil-ID teenuse töös.
MID_NOT_USER Telefoninumber ei kuulu Mobiil-ID kliendile.
VOTING_END Hääletusperiood on lõppenud.

Valijarakendus teeb päringu `RPC.Sign` hääle allkirjastamise algatamiseks.

params.AuthMethod Toetatud ainult autentimismeetod `ticket`.
params.AuthToken Mobiil-ID autentimistõend.
params.Hash BASE64-kodeeritud elektroonilise hääle SHA-256 räsi
params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.
params.PhoneNo Hääle allkirjastaja telefoninumber

```
{
    "id": 0.0,
    "method": "RPC.Sign",
    "params": [
        {
            "AuthMethod": "ticket",
            "AuthToken":
↪"G1RTZqBSBKrzqReuKYrmFUFXWFPvaxhJjdiZi6zqAnaK3OvrT...",
            "Hash": "9IBrA05ylt2StdjxKkSTYMW/rQXY3Vub4upzShdfEzo=",
            "OS": "Operating System,2,0",
            "PhoneNo": "+37200000766",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
        }
    ]
}
```

result.ChallengeID Mobiil-ID kontrollkood kuvamiseks valijarakenduses
result.SessionCode Mobiil-ID seansiidentifikaator edasiste poll-päringute jaoks.

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "ChallengeID": "7866",
        "SessionCode": "E663A711BB9447EAD82491F9372F4CA",
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
```

```
    }
}
```

Võimalikud veateated päringu `RPC.Sign` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
MID_BAD_CERTIFICATE Viga valija Mobiil-ID allkirjastamissertifikaadiga.
MID_NOT_USER Telefoninumber ei kuulu Mobiil-ID kliendile.
VOTING_END Hääletusperiood on lõppenud.

Valijarakendus teeb päringu `RPC.SignStatus` allkirjastamisprotsessi seisundi hindamiseks.

params.OS Operatsioonisüsteem, millel valijarakendust kasutatakse.
params.SessionCode Mobiil-ID seansiidentifikaator

```
{
    "id": 0.0,
    "method": "RPC.SignStatus",
    "params": [
        {
            "OS": "Operating System,2,0",
            "SessionCode": "E663A711BB9447EAD82491F9372F4CA",
            "SessionID": "057229fdfa2df7d3c7f4ced81b02760b"
        }
    ]
}
```

result.Signature Juhul kui vastuse Status väli on OK, BASE-64 kodeeritud PKCS1 vormingus signatuur, vastasel juhul `null`.
result.Status Päringu staatus - POLL viitab vajadusele päringut korrata, OK viitab edukale allkirjastamisele. Vastuse muud väljad sisaldavad infot vaid siis kui väärtus on OK.

```
{
    "error": null,
    "id": 0.0,
    "result": {
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b",
        "Signature": null,
        "Status": "POLL"
    }
}
```

```
{
    "error": null,
    "id": 0.0,
    "result": {
```

```
        "SessionID": "057229fdfa2df7d3c7f4ced81b02760b",
        "Signature": "MOj+8xQ9DmZPr/
↪ItHlm0tHNMCuTgn6dT9jcXjPLf0+2sVjsS11jRI...",
        "Status": "OK"
    }
}
```

Võimalikud veateated päringu `RPC.SignStatus` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
MID_ABSENT Valija mobiiltelefon ei ole kättesaadav.
MID_BAD_CERTIFICATE Viga valija Mobiil-ID allkirjastamissertifikaadiga.
MID_CANCELED Valija katkestas Mobiil-ID seansi.
MID_EXPIRED Mobiil-ID seanss on aegunud.
MID_GENERAL Viga Mobiil-ID teenuse töös.
VOTING_END Hääletusperiood on lõppenud.

7.5 Hääle kontrollimine

Kontrollrakendus teeb päringu `RPC.Verify` allkirjastatud hääle ning häält kvalifitseerivate tõendite allalaadimiseks kogumisteenusest.

params.OS Operatsioonisüsteem, millel kontrollrakendust kasutatakse.
params.VoteID QR-koodi vahendusel valijarakendusest saadud hääle identifikaator talletusteenuses.

```
{
    "id": 1,
    "method": "RPC.Verify",
    "params": [
        {
            "OS": "Operating System,2,0",
            "SessionID": "ec3a0cab353d552952289f2c7ad52e27",
            "VoteID": "VM/cUIU4n7VjxpUx1fC00Q=="
        }
    ]
}
```

result.Qualification.ocsp
result.Qualification.tspreg Vaata peatükki hääle verifitseerimisest
result.Type Allkirjastatud hääle vorming. Hetkel on ainus toetatud väärtus `bdoc`.
result.Vote BASE64-kodeeritud hääl `SignedVote` eelpoolmääratud vormingus.

```
{
    "error": null,
    "id": 1,
    "result": {
        "Qualification": {
            "ocsp":
↪"MIIG8woBAKCCBuwwggboBgkrBgEFBQcwAQEEggbZMIIG1TCCASehgY...",
            "tspreg":
↪"MIIE0QYJKoZIhvcNAQcCoIIEwjCCBL4CAQMxDzANBglghkgBDQEJE..."
        },
        "SessionID": "027ab451969d9d3f044ea2cb2675b503",
        "Type": "bdoc",
        "Vote":
↪"UEsDBAoAAAAAAAAAACKIflFHwAAAB8AAAAIAAAAbWltZXR5cGVhcHB..."
    }
}
```

Võimalikud veateated päringu `RPC.Verify` korral.

BAD_REQUEST Vigane päring.
INTERNAL_SERVER_ERROR Viga serveri sisemises töös.
VOTING_END Hääletusperiood on lõppenud.

PEATÜKK 8

E-urni töötlemine

8.1 Tühistus- ja ennistusnimekiri

Tühistus- ja ennistusnimekiri sisaldab andmeid isikute kohta, kelle e-hääl tuleb tühistada (ei lähe arvesse valimistulemuste kokkulugemisel) või ennistada (s.t. tühistatakse eelnev tühistamine ning häälte uuesti üle lugemisel võetakse ennistatud e-hääl arvesse). Nimekiri laaditakse süsteemi digitaalselt allkirjastatud dokumendina, mille andmefaili vorming on järgmine:

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",

    "definitions": {
        "rev_entry": {
            "type": "string",
            "pattern": "^[0-9]+.[0-9]+$"
        }
    },

    "type": "object",
    "properties": {
        "election": {"type": "string"},
        "type": {"enum": ["revoke", "restore"]},
        "persons": {
            "type": "array",
            "items": {
                "$ref": "#/definitions/rev_entry"
            }
        }
    },
    "required": ["election", "persons", "type"],
    "additionalProperties": false
}
```

Näide:

```
{
  "election": "TESTKOV",
  "persons": [
    "11412090004",
    "11412090005",
    "11412090006"
  ],
  "type": "revoke"
}
```

8.2 E-hääletanute nimekiri

E-hääletanute nimekiri on pärast e-hääletamise lõppu väljastatav nimekiri e-hääletanud isikutest, sorteerituna valimisjaoskondade kaupa. Dokument genereeritakse töötlemisrakenduse poolt.

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",

    "definitions": {
        "onlinevoters": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+$": {
                    "type": "array",
                    "items": [
                        {
                            "type": "string"
                        },
                        {
                            "type": "number"
                        }
                    ],
                    "additionalItems": false
                }
            },
            "additionalProperties": false
        },
        "stations": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/onlinevoters"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
```

```
        },
        "districts": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/stations"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
        }
    },

    "type": "object",
    "properties": {
        "election": {"type": "string"},
        "onlinevoters": {
            "$ref": "#/definitions/districts"
        }
    },
    "required": ["election", "onlinevoters"],
    "additionalProperties": false
}
```

Näide:

```
{
  "election": "TESTKOV",
  "onlinevoters": {
    "164.1": {
      "164.1": {
        "11412090001": [
          "Nimi Nimeste",
          1
        ],
        "11412090002": [
          "Nimi Nimeste",
          1
        ],
        "11412090003": [
          "Nimi Nimeste",
          1
        ]
      }
    },
    "296.1": {
      "296.1": {
        "11412090004": [
          "Nimi Nimeste",
          1
```

```
        ],
        "11412090005": [
          "Nimi Nimeste",
          1
        ],
        "11412090006": [
          "Nimi Nimeste",
          1
        ]
      },
      "296.2": {
        "11412090007": [
          "Nimi Nimeste",
          1
        ],
        "11412090008": [
          "Nimi Nimeste",
          1
        ],
        "11412090009": [
          "Nimi Nimeste",
          1
        ]
      }
    }
  }
}
```

8.3 Hääletamistulemus

Võtmerakenduse poolt dekrüpteeritud ning summeeritud hääled jagatud valimisringkondade ja jaoskondade kaupa.

Hääletamistulemuste failis peavad iga jaoskonna kohta olema järgmised andmed.

1. Rikutud ja kehtetute häälte arvu näitav kirje. Seda ka juhul, kui valimisjaoskonnas polnud ühtki rikutud või kehtetut häält: sellisel juhul on häälte arv null.
2. Iga valiku poolt antud häälte arvu näitav kirje. Seda ka juhul, kui valimisjaoskonnas ei antud selle valiku poolt ühtki häält: sellisel juhul on häälte arv null.

```
1  {
2      "$schema": "http://json-schema.org/draft-04/schema#",
3
4      "definitions": {
5          "results": {
6              "type": "object",
7              "properties": {
8                  "invalid": {
```

```
                    "type": "integer"
                }
            },
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "type": "integer"
                }
            },
            "additionalProperties": false,
            "required": ["invalid"]
        },
        "stations": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/results"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
        },
        "district_dict": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/results"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
        },
        "stations_dict": {
            "type": "object",
            "patternProperties": {
                "^[0-9]+.[0-9]+$": {
                    "$ref": "#/definitions/stations"
                }
            },
            "additionalProperties": false,
            "minProperties": 1
        }
    },

    "type": "object",
    "properties": {
        "election": {"type": "string"},
        "bydistrict": {
            "$ref": "#/definitions/district_dict"
        },
        "bystation": {
```

```
            "$ref": "#/definitions/stations_dict"
        }
    },
    "required": ["election", "bydistrict", "bystation"],
    "additionalProperties": false
}
```

Näide:

```
{
  "bydistrict": {
    "164.1": {
      "164.126": 0,
      "164.127": 0,
      "invalid": 0
    },
    "296.1": {
      "296.101": 0,
      "296.102": 0,
      "296.115": 0,
      "296.116": 0,
      "296.117": 0,
      "296.198": 0,
      "296.199": 0,
      "296.200": 0,
      "invalid": 0
    }
  },
  "bystation": {
    "164.1": {
      "164.1": {
        "164.126": 0,
        "164.127": 0,
        "invalid": 0
      }
    },
    "296.1": {
      "296.1": {
        "296.101": 0,
        "296.102": 0,
        "296.115": 0,
        "296.116": 0,
        "296.117": 0,
        "296.198": 0,
        "296.199": 0,
        "296.200": 0,
        "invalid": 0
      },
      "296.2": {
        "296.101": 0,
```

```
        "296.102": 0,
        "296.115": 0,
        "296.116": 0,
        "296.117": 0,
        "296.198": 0,
        "296.199": 0,
        "296.200": 0,
        "invalid": 0
      }
    },
  },
  "election": "TESTKOV"
}
```

8.4 E-urn

Fail sisaldab kogumisteenuse poolt vastu võetud hääli koos häälte juurde kuuluvate andmetega.

Faili vorming on Zip64 konteiner.

Valija-spetsiifilised kaustad asuvad vahetult juurkausta all, fikseeritud nimega ülemkausta pakis ei ole.

Faili sisu:

- `<voter id>/`
- `<timestamp>.version`
- `<timestamp>.<vote type>`
- `<timestamp>.<qualifier>*`

kus:

- `<voter id>` on valija identifikaator, Eesti puhul isikukood;
- `<timestamp>` on hääle esitamise kellaaeg vormingus `yyyymmddhhmmss±zzzz`;
  - see kellaaeg kajastab hetke, mil päring kogumisteenusesse tehti, ja on antud lihtsalt urni inimloetavuse parandamiseks; hääle tegelik ajamärk või -tempel on mõne kvalifitseeriva vastuse sees;
- `<vote type>` on valikute konteineri tüüp, Eesti puhul BDOC;
  - kusjuures BDOC ise on lihtsalt põhiprofiiliga ja ei sisalda kvalifitseerivad parameetreid (kehtivuskinnitusi, ajamärgendeid, ajatempleid),
- `<qualifier>` on häält kvalifitseeriva protokolli tüüp, millest hetkel võimalikud on:
  - `ocsp` - *Online Certificate Status Protocol* (kehtivuskinnitus, RFC 6960) kinnitab valija allkirjastamissertifikaadi kehtivust hääle andmise hetkel,

  - `ocsptm` - sama, mis `ocsp`, aga kasutab BDOC spetsifikatsiooni jaotises 6.1 kirjeldatud laiendust, kus nonsiks pannakse hääle allkirja räsi, et häält ajamärgendada,
  - `tsp` - *Time-Stamp Protocol* (ajatempel, RFC 3161) kinnitab, et päringu tegemise hetkeks oli hääl olemas,
  - `tspreg` - sama, mis `tsp`, aga nonsiks pannakse kogumisteenuse allkiri päringu `MessageImprint` elemendil, et häält registreerida.
- Iga hääle kohta esinevad failid on:
- `<timestamp>.version` - hääle andmise ajal kehtinud valijate nimekirja versioon;
- `<timestamp>.<vote type>` - valikute konteiner, mille sees on valiku identifikaator kujul `<valimise id>.<küsimuse id>.ballot`. Eesti puhul BDOC-konteineris olev vastava nimega fail;
- `<timestamp>.<qualifier>` - häält kvalifitseeriva protokolli päringu vastus; neid võib esineda mitu, aga iga iga protokolli kohta maksimaalselt üks.

PEATÜKK 9

Hääletamistulemuse audit

9.1 Miksimistõendi kontroll

Miksimistõendi kontrollimiseks kasutatakse algoritmi nagu on defineeritud Verificatumi verifitseerija implementeerimise manuaalis.

Märgime, et miksimistõendi koostamisel lisatakse krüptogrammile andmed valimiste, ringkonna, jaoskonna ja küsimuse identifikaatori kohta. Lisamiseks kodeeritakse vastav väli rühma elemendina, kasutades pimendamiseks juhuslikkust 0. Näitena, kui esialgu on krüptogramm $c_0 = (c_0 0, c_0 1)$, kasutades avalikku võtit $pk = (g, y)$, siis Verificatumi sisendina kasutatakse laia krüptogrammi $C = (c_{id}, c_d, c_s, c_q, c_0)$, kus:

- valimiste identifikaatori pseudokrüptogramm on antud kujul $c_{id} = (1, encode(id))$, kus funktsioon $encode$ kodeerib sõne vastava rühma elemendina ja *id* on valimiste identifikaatori sõne.
- ringkonna identifikaatori pseudokrüptogramm on antud kujul $c_d = (1, encode(d))$, kus *d* on ringkonna identifikaatori sõne.
- jaoskonna identifikaatori pseudokrüptogramm on antud kujul $c_s = (1, encode(s))$, kus *s* on jaoskonna identifikaatori sõne.
- küsimuse identifikaatori pseudokrüptogramm on antud kujul $c_q = (1, encode(q))$, kus *q* on küsimuse identifikaatori sõne.

Sellisel juhul defineeritakse laia krüptogrammile vastava avaliku võtmena $((g, 1), (g, 1), (g, 1), (g, 1), (g, y))$.

9.2 Korrektse dekrüpteerimise tõendi kontroll

Olgu antud krüptogramm $c = (c_0, c_1)$, mis deküpteeritakse väärtuseks d antud avaliku võtmega pk üle parameetrite (p, g) ja dekrüpteerimistõendiga (a, b, s).

Korrektse dekrüpteerimise kontrollimise jaoks arvutatakse väljakutse $k = H(”DECRYPTION|pk||c||d||a||b)$, kus H on SHA-256 räsifunktsioon. Seejärel kontrollitakse, et $c_0^s = a * (c_1/d)^k$ ja $g^s = b * y^k$.

9.3 Korrektse teisendamise kontroll

Kontrollimaks, et teisendus IVXV e-urni ja Verificatumi krüptogrammide vahel on tehtud korrektselt, tuleb korrata teisendust sõltumatult. Pärast sõltumatut teisendust tuleb võrrelda saadud väljundeid. Kuna teisendamine on deterministlik protseduur, siis garanteerib kordamine tegevuse õigsuse.