Kanta FHIR sähköinen allekirjoitus

v. 1.1.0

29.2.2024

Kanta-palvelut

Julkinen

Kela
Kanta-palvelut
PL 450, 00056 Kela

FPA
Kanta tjänsterma
PB 450, 00056 FPA

www.kanta.fi
kanta@kanta.fi

Muutoshistoria

Versio	Muutos	Tekijä	PVM
0.8	Ensimmäinen julkaistava draft-versio kommentointia varten	Kanta-palvelut	23.10.2023
1.0.0	Ensimmäinen virallinen julkaisu	Kanta-palvelut	20.12.2023
1.1.0	Poistettu lukujen 1.3 ja 3.1 teksti pitkäaikaisten allekirjoitusten tukemisesta ja päivitetty lukuun 3.1 oikea profiili JAdES-B-B	Kanta-palvelut	29.02.2024

Sisällys

1 Johdanto

FHIR (Fast Healthcare Interoperability Resources) on HL7:n (Health Level Seven) julkaisema datarakennestandardi, joka määrittelee, miten terveystietoja voidaan lähettää ja jakaa eri IT-järjestelmien välillä. FHIR-standardi edistää terveydenhuollon tietojen yhteensopivuutta ja saatavuutta, mahdollistaen tehokkaamman tiedonvaihdon eri terveydenhuollon toimijoiden välillä.

FHIR-resurssit tulee allekirjoittaa, jotta voidaan varmistua, että FHIR-data säilyy muuttumattomana ja että data on aitoa ja peräisin oikealta lähettäjältä. Resursseille tehtävä JWS-allekirjoitus perustuu digitaaliseen allekirjoitukseen, joka varmentaa tiedon eheyden ja lähettäjän identiteetin.

Tämä dokumentti tarjoaa ohjeet JWS-allekirjoituksen luomiseen FHIR-resursseille sekä esittää keinot allekirjoituksen tarkastamiseksi. Näin varmistetaan, että FHIR-muotoisten terveystietojen vaihto tapahtuu luotettavasti ja turvallisesti eri toimijoiden välillä.

1.1 Dokumentin rakenne

Tämä dokumentaatio määrittelee tavan luoda ja lisätä digitaalinen allekirjoitus HL7 FHIR Specification (v4.0.1: R4 - STU) mukaisiin sähköisiin resursseihin. Tätä voidaan jatkossa soveltaa myös HL7 FHIR Specification (v5.0.0: R5 - STU) mukaisiin resursseihin.

Dokumentti määrittelee FHIR-resurssien sähköisiin allekirjoituksiin käytettävät käytännöt ja toteutuksen.

Dokumentti määrittelee JSON Digital Signatures [5] tyyppisen allekirjoituksen käytön HL7 FHIR-resurssien digitaaliseen allekirjoitukseen [1] FHIR Specification (v4.0.1: R4 - STU) 6.1.2 Digital Signatures.

Dokumentissa käsitellään vain JSON Digital Signatures [5] tyyppisen sähköisen allekirjoituksen käyttö FHIR JSON (FHIR Specification (v4.0.1: R4 - STU ja v5.0.0: R5 - STU) resurssien allekirjoitukseen.

Luvut 1, 3 ja 5 ovat normatiivisia. Luku 2 käsittelee yleisemmin FHIR-rakennetta ja luku 3 määrittää allekirjoitusrakennetta koskevat asiat. Luku 4 sisältää konkreettisia esimerkkejä allekirjoituksen luonnista, sen tarkastuksesta ja selitykset JWS otsikoille. Luku 5 sisältää normatiivisen kuvauksen allekirjoitusprosessista.

1.2 Sähköisen allekirjoituksen yleiset periaatteet

Yleisellä tasolla digitaaliset allekirjoitukset toimivat siten, että allekirjoituksen luoja luo julkisen ja yksityisen avaimen parin ja jakaa julkisen avaimensa allekirjoitetun resurssin vastaanottajalle. Allekirjoitus perustuu resurssin tiivisteen (hash) laskemiseen ja yksityisellä avaimella

salaamiseen. Jos allekirjoitettava data on JSON-formaatissa, se tulee myös kanonikalisoida eli yhtenäistää ennen tiivisteen laskemista.

Resurssin vastaanottaja käyttää julkista avainta purkaakseen allekirjoituksen salauksen ja tarkistaakseen tiivisteen eheyden. Tämän jälkeen vastaanottaja laskee resurssista tiivisteen ja vertaa sitä tiivisteeseen, jonka hän sai purkaessaan allekirjoitusta. Jos tiivisteet ovat identtisiä, allekirjoitus on validi eli resurssi ei ole muuttunut allekirjoituksen jälkeen.

1.3 Sähköiset FHIR-resurssien allekirjoitukset Kanta-palveluissa

Digitaaliset allekirjoitukset ovat eräs sähköisen allekirjoituksen muoto. Ne käyttävät salaustekniikkaa ja sertifiointiviranomaisen (Certification Authority, CA) myöntämää varmennetta (digital certificate). CA:n myöntämä varmenne varmistaa, että allekirjoituksen aitouteen ja kiistämättömyyteen voi luottaa.

Digitaaliset allekirjoitukset täyttävät seuraavat toiminnalliset vaatimukset:

- Autentikaatio – Niillä voi varmistaa allekirjoittajan identiteetin
- Eheyden varmistaminen – Niillä voidaan taata, että allekirjoitettua asiakirjaa ei ole muutettu.
- Kiistämättömyys – Allekirjoittaja ei voi kiistää allekirjoituksen tekemistä.

Kanta FHIR-allekirjoituksessa täytyy käyttää SOTE järjestelmäallekirjoitusta.

HL7 FHIR resursseja (resources) allekirjoitetaan lisäämällä Bundle-resurssiin irrotettu (detached) JSON signature.

- JSON-digitaaliallekirjoituksen täytyy noudattaa kappaleessa 1.5.1 viitattuja standardeja ja spesifikaatioita (JAdES, FHIR, RFC).
- JSON-digitaaliallekirjoituksen täytyy noudattaa JAdES-standardia.

1.4 Dokumentissa käytetyt termit, symbolit ja lyhenteet

1.4.1 Lyhenteet

Lyhenne	Selite
JSON	JavaScript Object Notation
JWS	JSON Web Signature
JOSE	JSON Object Signing and Encryption
OCSP	Online Certificate Status Protocol
SHA	Secure Hash Algorithm
URI	Uniform Resource Identifier

URL	Uniform Resource Locator
URN	Uniform Resource Name
UTC	Coordinated Universal Time
JCS	JSON Canonicalization Scheme

Taulukko 1: Käytetyt lyhenteet

1.4.2 Terminologia

Tässä asiakirjassa käytetään aina, kun mahdollista, samaa terminologiaa kuin IETF RFC 7515:ssä [2] ja IETF RFC 8259:ssä [1].

Tässä asiakirjassa termi "JSON Web Signature" tarkoittaa IETF RFC 7515:ssä [2] määriteltyä JSON-allekirjoitusrakennetta.

Tässä asiakirjassa käytetään termiä "JSON-arvo" ilmaisemaan JSON-objekteja, JSON-taulukoita tai JSON-numeroita tai JSON-merkkijonoja, eli osajoukkoa "JSON-arvon" mahdollisista merkityksistä, jotka on lueteltu IETF RFC 8259:n [1] lausekkeessa 3. .

Tässä asiakirjassa käytetään termiä "otsikkoparametri" kuvaamaan JSON-objektia, JSON-taulukkoa, JSON-numeroa tai JSON-merkkijonoa, joka on joko IETF RFC 8259:ssä [1] määritellyn JWS Protected Headerin tai JWS:n suojaamattoman otsikon jäsen.

Tässä asiakirjassa käytetään termiä "jäsen" ilmaisemaan JSON-objektin jäsen IETF RFC 8259:n [1] lausekkeen 4 mukaisesti.

Tässä asiakirjassa käytetään termiä "elementti" tai "taulukon elementti" osoittamaan JSON-taulukon sijainnin sisältöä (määritetty IETF RFC 8259:n [1] lausekkeessa 5).

Tässä asiakirjassa käytetään termiä "JCS" ilmaisemaan standardoitu tapa kuvata JSON dokumentti kanonikalisoidulla muodolla.

Tässä asiakirjassa käytetään termiä "Täytyy" ilmaisemaan ehdotonta vaatimusta kaikille toteutuksille.

Tässä asiakirjassa käytetään termiä "Pitäisi" ilmaisemaan paras käytäntö tai suositus, joka on otettava huomioon toteutuksen yhteydessä. Saattaa olla päteviä syitä jättää huomioimatta jokin kohde, mutta täytyy ymmärtää ja punnita huolellisesti kaikki seuraukset ennen kuin valitaan toinen vaihtoehto.

Tässä asiakirjassa käytetään termiä "Voi" ilmaisemaan valinnaisuus toteutuksessa. Kohde voidaan sisällyttää tai jättää pois toteuttajan päätöksen mukaan ilman seuraamuksia.

Tässä asiakirjassa käytetään termiä "Ei saa" ilmaisemaan ehdoton kielto kaikille toteutuksille.

1.5 Viittaukset

Viittaukset lähteisiin ovat joko erityisiä (julkaisupäivän ja/tai painoksen tai versionumeron perusteella) tai epäspesifisiä. Tiettyihin viittauksiin sovelletaan vain lainattua versiota. Epäspesifisiin viittauksiin sovelletaan viitatun asiakirjan viimeisintä versiota (mukaan lukien mahdolliset muutokset).

1.5.1 Normatiiviset viittaukset

[1] FHIR Specification (v4.0.1: R4 - STU): http://hl7.org/fhir/R4/index.html

[2] FHIR Specification (v5.0.0: R5 - STU): http://hl7.org/fhir/R5/index.html

[3] ETSI TS 119 182-1 V1.1.1 (2021-03): Electronic Signatures and Infrastructures (ESI); JAdES digital signatures; Technical Specification, https://www.etsi.org/deliver/etsi_ts/119100_119199/11918201/01.01.01_60/ts_11918201v010101p.pdf

[4] IETF RFC 8259 (December 2017): "The JavaScript Object Notation (JSON) Data Interchange Format", https://www.rfc-editor.org/rfc/rfc8259.html

[5] IETF RFC 7515 (May 2015): "JSON Web Signature (JWS)", https://www.rfc-editor.org/rfc/rfc7515.html

[6] IETF RFC 7519 (May 2015): JSON Web Token (JWT), https://www.rfc-editor.org/rfc/rfc7519.html

[7] Kyberturvallisuuskeskus Kryptograafiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen, https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/file/ohje-kryptografiset-vahvuusvaatimukset-kansalliset-suojaustasot.pdf

[8] DVV Palveluvarmenteet sosiaali- ja terveydenhuollolle, https://dvv.fi/palveluvarmenteet-sosiaali-ja-terveydenhuollolle

[9] IETF RFC 8785 (June 2020): JSON Canonicalization Scheme (JCS), https://www.rfc-editor.org/rfc/rfc8785.html

[10] IETF RFC 3986 (January 2005): Uniform Resource Identifier (URI): https://www.rfc-editor.org/rfc/rfc3986.html

[11] ETSI:n TS 119 312 V1.2.1 (2017-05): Electronic Signatures and Infrastructures (ESI); Cryptographic Suites, https://www.etsi.org/deliver/etsi_ts/119300_119399/119312/01.02.01_60/ts_119312v010201p.pdf

[12] ECMA-262, 10th edition, June 2019 ECMAScript® 2019 Language Specification https://262.ecma-international.org/10.0/

1.5.2 Informatiiviset viittaukset

[i.1] https://www.kanta.fi/jarjestelmakehittajat/sahkoisen-allekirjoituksen-maarittely

[i.2] https://www.rfc-editor.org/rfc/rfc6901

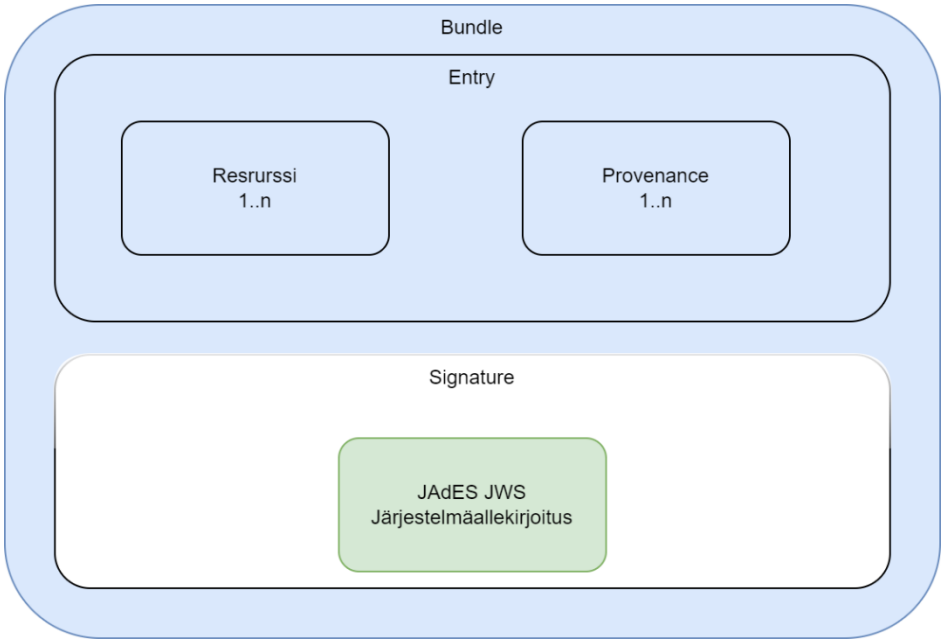
[i.3] DSS (Digital Signature Service) version : 5.12.1 - 2023-03-15, https://ec.europa.eu/digital-building-blocks/wikis/display/DIGITAL/Digital+Signature+Service+-++DSS

[i.4] https://hl7.org/fhir/R4/datatypes-definitions.html#signature

2 FHIR-resurssien sähköinen allekirjoitus

2.1 FHIR resurssien allekirjoituksen rakenne

Tämä kappale sisältää esimerkin FHIR spesifikaation mukaisista JSON resurssista. Kanta-FHIR-resurssien allekirjoituksissa tuetaan ensimmäisessä vaiheessa koko Bundle-resurssin allekirjoitusta, jolloin Signature-elementti sijaitsee Bundle-resurssissa. Tulevaisuudessa uusien käyttötarpeiden osalta (useampi allekirjoitus, moniallekirjoitus usealle yksittäiselle resurssille resurssikohtaisesti) määrittelyitä täydennetään tämän dokumentin uusiin versioihin.



Kuva 1: FHIR Bundlen rakenne allekirjoituksen kanssa

FHIR Signature on JSON-elementti, johon JWS allekirjoitus sisällytetään. Kanta JWS allekirjoitukset tehdään FHIR Bundle-resursseista. Luotu allekirjoitus lisätään Signature-elementin data-kenttään.

Allekirjoituksen kohdistumisesta puhutaan luvussa 4.4.

```
{
  // from Element: extension
  "type" : [{ Coding }], // Syy allekirjoitukselle
  "when" : "<instant>", // Allekirjoituksen ajankohta
  "who" : { Reference(Device|Organization|Patient|Practitioner| PractitionerRole|Related-
Person) }, // Allekirjoituksen luoja
  "onBehalfOf" : { Reference(Device|Organization|Patient|Practitioner|Practitioner-
Role|RelatedPerson) }, // Allekirjoitusta edustava taho
  "targetFormat" : "<code>", // Allekirjoitettavien resurssien tekninen formaatti
  "sigFormat" : "<code>", // Allekirjoituksen tekninen formaatti
  "data" : "<base64Binary>" // Allekirjoitus merkkijonona
}
```

Esimerkki 1: FHIR Signature-elementin rakenne

Yllä on esitetty FHIR R4 -standardin mukainen Signature -tyyppisen elementin schema [i4] FHIR Specification (v4.0.1: R4 - STU) 2.24.0.17 Signature Yllä olevassa skeemassa FHIR Specification (v4.0.1: R4 - STU) pakollisia kenttiä ovat type, when ja who, muut Signature elementin kentät ovat optionaalisia. Standardin R5-versiossa on väljennetty kenttien pakollisuutta. Allekirjoituksen kenttien pakollisuuteen voidaan ottaa kantaa myös Kanta-palveluiden profiloinnilla.

JWS allekirjoitus sisällytetään Signature-elementin data-parametriin BASE64URL enkoodattuna ilman payload-osaa muodossa "BASE64URL(UTF8(JWS Protected Header))..BASE64URL(JWS Signature)".

Signature-elementissä:

- type = Lisätietoja allekirjoituksesta. Typessä voidaan käyttää organisaatiotunnistetta.
  - system = Viittaa johonkin standardiin. (ASTM standardi esimerkissä).
  - code = Tunniste, jolle on selitys FHIR standardissa [1] FHIR Specification (v4.0.1: R4 - STU). Täytyy sisältää saman tiedon kuin srCms parametri JWS otsikossa.
  - display = Sanallinen selitys mistä on kyse.
- when = Ajankohta allekirjoituksen luonnille UTC-formaatissa.
- who = Allekirjoituksen luojan tunniste.
- targetFormat = Allekirjoitettujen resurssien tekninen formaatti. Tämän täytyy olla aina application/fhir+json.
- sigFormat = Allekirjoituksen tekninen formaatti. Tämän täytyy olla aina application/jose.
- data = JWS allekirjoitus. RFC4648 mukaisesti BASE64-enkoodattuna

```
"signature" : {
  "type" : [{
    "system" : "urn:iso-astm:E1762-95:2013",
    "code" : "  1.2.840.10065.1.12.1.13",
    "display" : "Review Signature"
  }],
  "when" : "2022-02-08T10:16:32.000+10:00",
  "who": {
      "identifier" : {
           "system" : "urn:ietf:rfc:3986",
           "value" : "urn:oid:1.2.246.xxxx.yyyy.zzz"
    },
    "display" : "Organisaation nimi"
  },
  "targetFormat" : "application/fhir+json",
  "sigFormat" : "application/jose",
  "data" : "UEQ5NGJXd2dkbVZ5YzJsdmJqMGlNUzR3SWlCbGJt
TnZaR2x1WnowaVZWUkdMVGdpLi4xY21VK0Nqd3ZSVzUyWld4dmNHVStJQT09"
}
```

Esimerkki 2: FHIR-signature-elementti [i.4] Esimerkki Signature-elementistä, joka sisältää JWS-allekirjoituksen

2.2 FHIR-resurssin allekirjoittaja who-rakenteessa

Bundlen allekirjoittaja ilmaistaan who-kentässä viittauksella käyttäen loogista viittausta ja display-arvoa. Koska allekirjoitus tehdään järjestelmävarmenteella, joka on lähtökohtaisesti liittyjäkohtainen, on allekirjoittajana liittyjäorganisaatio. Jos allekirjoituksen tekee muu taho kuin liittyjäorganisaatio, annetaan tiedoissa allekirjoittavan vastuuorganisaation (Kanta-valittäjä) tunniste.

```
"who": {
      "identifier" : {
           "system" : "urn:ietf:rfc:3986",
           "value" : "urn:oid:1.2.246.xxxx.yyyy.zzz"
    },
    "display" : "Organisaation nimi"
}
```

3 Sähköisen allekirjoituksen vaatimukset Kanta-palveluissa

3.1 Sähköisiä allekirjoituksia koskevat sitovat vaatimukset

[1] FHIR Specification (v4.0.1: R4 - STU) 2.24.0.17 Signature mukaiset vaatimukset digitaalisille allekirjoituksille:

1. Signature.data on base64url enkoodattu JWS-allekirjoitus [5] IETF RFC 7515 (May 2015)

2. Allekirjoitus on irrallinen (missä allekirjoitus on irrallaan itse allekirjoitettavasta sisällöstä)
3. Kun FHIR resurssi on allekirjoitettu, allekirjoitus on tehty resurssin kanonikalisessa JSON-muodossa. Kanonikalisointivaatimukset on esitetty luvussa 4.5.
4. Allekirjoituksen pitäisi käyttää minimissään tiivistealgoritmia SHA256. Allekirjoituksen vahvistuskäytäntö koskee allekirjoitusta ja määrittää hyväksyttävyyden.
5. Allekirjoitus täytyy sisältää "srCms signer commitments" -elementin allekirjoituksen "tarkoitusta" varten. Tarkoitus voi olla todistettu toiminto tai allekirjoitukseen liittyvä rooli. Arvon tulee olla standardista ASTM E1762-95(2013). Signature.type sisältää samat arvot kuin srCms-elementti.
6. Allekirjoituksen täytyy olla JAdES-B-B -profiilin mukainen.

Vaatimukset allekirjoituksen verifioinnille [1] FHIR Specification (v4.0.1: R4 - STU) 2.1.28.0.18.2 JSON Signature rules:

1. Tarkastaa, että digitaalisen allekirjoituksen lohko on eheä JWS-allekirjoituksen kautta
2. Vahvistaa, että allekirjoittaja on aito, voimassa ja sopii allekirjoituksen tarkoitukseen
3. Vahvistaa, että allekirjoitettava sisältö on muokkaamaton käyttäen hash algoritmia

Allekirjoitusten kanonikalisointi vaatimukset [1] FHIR Specification (v4.0.1: R4 - STU) 2.1.6.4.5 Canonical JSON:

Allekirjoitettu fragmentti kanonikalisoidaan siten, että:

1. Ei tyhjiä rivejä muuta kuin yksittäisiä välilyöntejä
2. Muuttujat on järjestetty aakkosjärjestyksessä kunkin objektin sisällä
3. Useita fragmentteja ketjutetaan ilman välilyöntejä Signature-tietotyypin elementin määrittämässä järjestyksessä

3.1.2 Käytetyt tiivistealgoritmit

Liikenne- ja viestintävirasto Traficomin/Kyberturvallisuuskeskus ohje antaa [7] Kyberturvallisuuskeskus Kryptograafiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen sisältää kryptografiset vähimmäisvaatimukset turvallisuusluokitellun tiedon suojaamiseen. Ohje määrittelee digitaalisia allekirjoituksia varten tason 4 (TL4) dokumenteissa käytettävät tiivistealgoritmit. FHIR resurssit on luokiteltu turvallisuustason TL4 luokkaan.

JAdES standardi [3] ETSI TS 119 182-1 V1.1.1 (2021-03): Electronic Signatures and Infrastructures (ESI) määrittelee myös allekirjoituksiin käytettävät kryptografiset tiivistealgoritmit

kohdassa Annex E (normative): Digest algorithms identifiers for JAdES signatures. JAdES standardin mukaan kryptografiset vahvuusvaatimukset voidaan korvata kansallisilla vaatimuksilla.

FHIR asiakirjojen allekirjoituksissa täytyy käyttää tiivisteiden laskemisessa ohjeen [7] Kyberturvallisuuskeskus Kryptograafiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen mukaisia mainittuja kansalliset määräykset turvallisuustasolla IV täyttäviä minimivaatimuksia tiivistealgoritmeille.

Kanta-palveluiden FHIR-allekirjoituksissa täytyy käyttää alla olevassa taulukossa listattua tuettua tiivistealgoritmia:

Algoritmi	Tarkenne/Kuvaus
SHA-2: SHA-256	Minimivaatimus Kansallinen turvallisuusluokka/kryptovahvuus TL IV Tiivistefunktio, käyttötarkoitus digitaaliset allekirjoitukset ja "hash-only" -sovellukset.
SHA-2: SHA-384	Tiivistefunktio, käyttötarkoitus digitaaliset allekirjoitukset ja "hash-only" -sovellukset.
SHA-2: SHA-512	Tiivistefunktio, käyttötarkoitus digitaaliset allekirjoitukset ja "hash-only" -sovellukset.

Taulukko 2: käytettävät tiivistealgoritmit

3.1.3 Allekirjoitusalgoritmit ja menetelmät

Liikenne- ja viestintävirasto Traficomin/Kyberturvallisuuskeskus ohje [7] Kyberturvallisuuskeskus Kryptograafiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen sisältää kryptografiset vähimmäisvaatimukset turvallisuusluokitellun tiedon suojaamiseen. Ohje määrittelee digitaalisia allekirjoituksia varten tason 4 (TL4) dokumenteissa käytettävät allekirjoitusalgoritmit. Kanta-palveluissa käsiteltävät FHIR-resurssit on luokiteltu turvallisuustason TL4 luokkaan.

JAdES standardi [3] ETSI TS 119 182-1 V1.1.1 (2021-03): Electronic Signatures and Infrastructures (ESI) määrittelee, että digitaalisten allekirjoitusten luomiseen ja täydentämiseen käytettävien algoritmit ja avainten pituudet tulee olla standardin [11] ETSI TS 119 312 V1.2.1 (2017-05) mukaisia. Lisäksi JAdES standardin mukaan [11] ETSI TS 119 312 V1.2.1 (2017-05) standardissa määritellyt kryptografiset vahvuusvaatimukset voidaan korvata kansallisilla vaatimuksilla.

FHIR resurssien allekirjoituksissa täytyy käyttää ohjeen [7] Kyberturvallisuuskeskus Kryptograafiset vahvuusvaatimukset luottamuksellisuuden suojaamiseen ja Digi- ja väestötietovirasto (DVV) ohjeen [8] DVV Palveluvarmenteet sosiaali- ja terveydenhuollolle -ohjeen mukaisia ja tuettuja SOTE-varmenteita.

Kanta-palveluiden FHIR-allekirjoituksissa täytyy käyttää alla olevassa taulukossa listattua tuettua allekirjoitusalgoritmia:

Algoritmi (Äärellisen kunnan koko)	Tarkenne/Kuvaus
ECDSA[256]	**Minimivaatimus** Kansallinen turvallisuusluokka/kryptovahvuus TL IV Elliptic Curve Digital Signature Algorithm, allekirjoitus epäsymmetrisellä menetelmällä (esim. RSA) - "vaativat allekirjoitukset" Tuettu algoritmi ECDSA ES256 (EC P-256 DSA with SHA-256)
ECDSA[384]	Elliptic Curve Digital Signature Algorithm, allekirjoitus epäsymmetrisellä menetelmällä (esim. RSA) - "vaativat allekirjoitukset" Tuettu algoritmi ECDSA ES384 (EC P-384 DSA with SHA-384)
RSA[3072]	**Minimivaatimus** Kansallinen turvallisuusluokka/kryptovahvuus TL IV Allekirjoitus epäsymmetrisellä menetelmällä (esim. RSA) - "vaativat allekirjoitukset" Tuettu algoritmi RS256, RS384 ja RS512 (RSASSA-PKCS1-v1_5)
RSA[4096]	Allekirjoitus epäsymmetrisellä menetelmällä (esim. RSA) - "vaativat allekirjoitukset" Tuettu algoritmi RS256, RS384 ja RS512 (RSASSA-PKCS1-v1_5)

Taulukko 3: Allekirjoitusalgoritmit

4 JWS allekirjoitus

4.1 JWS allekirjoitus

JSON Web Signature (JWS) on IETF:n ehdottama standardi (RFC 7515) kaikenlaisen tiedon allekirjoittamiseen. HL7 FHIR standardi suosittelee JWS signaturen käyttöä FHIR resurssien digitaalisiin allekirjoituksiin.

HL7 FHIR allekirjoituksissa JWS on JSON compact serialization -muodossa, joka sisältää seuraavat osat, mutta ei hyötykuormaa:

- Otsikko (Header), joka sisältää arvon BASE64URL(UTF8(JWS Protected Header))
- Allekirjoitus (Signature), joka sisältää arvon BASE64URL(JWS Signature)

JWS otsikko (JWS Protected Header) sisältää JSON elementtejä, jotka ilmaisevat allekirjoitukseen käytetyt parametrit ja tiedot allekirjoitettavasta sisällöstä ja/tai sisällöstä muodostetut tiivisteet.

Arvojen otsikko, tyhjällä merkkijonolla korvattu data-osio ja allekirjoitus ketjutettuna järjestyksessä pisteillä ('.') osien välissä tuottaa serialisoidun JWS irroitetun (detached) allekirjoituksen.

HL7 FHIR allekirjoituksissa JWS hyötykuormaa (payload) ei sisällytetä serialisoituun esitys- ja tallennusmuotoon.

```
eyJ0eXAiOiJKV1QiLA0KICJhbGciOiJIUzI1NiJ9..dBjftJeZ4CVPmB92K27uhbUJU1p1r_wW1gFWFOEjXk
```

Taulukko 4: esimerkki JWS-allekirjoituksesta

4.2 FHIR Bundlen allekirjoitus

Kanta-palveluissa FHIR-resurssit allekirjoitetaan allekirjoittamalla Bundle, jossa ne siirretään ja tallennetaan.

JWS:n otsikon täytyy sisältää alg, typ, b64, crit, x5c, sigT ja sigD otsikkoparametrit.

SigD otsikkoparametri on JSON-objekti, jonka täytyy sisältää mId ja ctys jäsenet.

srCms otsikkoparametri on JSON-objekti, jonka täytyy sisältää commId jäsen ja voi sisältää commQuals jäsenen. FHIR spesifikaation [1] FHIR Specification (v4.0.1: R4 - STU) mukaan allekirjoituksen täytyy sisältää srCms-elementti.

```
{
  "alg": "<Allekirjoitusalgoritmi esim. RS256, RS384, RS512, ECDSA P-256 tai
ECDSA P-384>",
  "typ": "<JOSE, jos JWS or JSON signature>"
  // b64 on false, kun mId = ObjectIdByURIHash ja puuttuu tai true kun mId = Ob-
jectIdByURI
  "b64": true,
  // crit parametrissa tulee olla kaikki JWS/JAdES allekirjoituksen parametrit
  "crit": [{"alg","b64","typ","x5c","sigT","sigD","srCms"}],
  "x5c": [
     "<X.509 sertifikaatin julkinen avain>"
  ],
  "srCms": [
    {
      "commId": "<oId yksilöi allekirjoittajan tekemän sitoumuksen>",
      "commQuals": "<sitoumusta koskevat lisätiedot. Taulukko objekteja>"
    }
  ],
  "sigT": "<Allekirjoituksen aikaleima UTC muodossa>",
  "sigD": {
      // Jos mId = http://uri.etsi.org/19182/ObjectIdByURI, HashM ja HashV ei
tule headeriin, ctys sisältää payloadin sisältötyyppin (content type)
      "mId": "<URI identifioi hyötykuormaan viittauksiin käytetyn mekanismin>",
      "ctys": [
        "<Hyötykuorman sisältötyyppi (content type)>"
      ]
  }
}
.
{
  "<Hyötykuorma (payload)>"
}
.
{
```

```
 "Allekirjoitus" // BASE64URL(JWS signature)
}
```

Yllä on esitelty JWS:n rakenne, kun allekirjoitus kohdistuu Bundle-resurssiin. Allekirjoitettava Bundle-resurssi sisällytetään allekirjoituksen luonnissa JWS:n payload-elementtiin kanonisoituna ja ilman signature-elementtiä.

4.3 Allekirjoituksen kohdistaminen

FHIR allekirjoitus kohdistetaan koko Bundle-resurssiin, jossa Signature-elementti sijaitsee.

JAdES standardin [3] ETSI TS 119 182-1 V1.1.1 (2021-03) 5.8.2.3.1 General requirements -lauseke määrittelee mekanismeja (mechanism Id), jotka käyttävät URI:ita viitattaessa JWS-hyötykuorman rakentamiseen osallistuviin tieto-objekteihin.

FHIR Bundle-resurssin allekirjoitukseen käytetään mekanismia "http://uri.etsi.org/19182/ObjectIdByURI" (sigD-elementin mId-parametri). Allekirjoitettava bundle on aina JWS hyötykuormassa (payload), eli viittauksia ei käytetä.

4.4 Allekirjoitettavan tiedon kanonikalisointi

Kryptograafiset operaatiot, kuten tiivistäminen (hashaus) ja allekirjoittaminen, edellyttävät, että allekirjoitettava sisältö säilyy muuttumattomana koko käsittelyn ajan. JSON-kanonikalisointi eli JCS (JSON Canonicalization Scheme) on menetelmä, joka varmistaa yhtenäisen esityksen JSON-rakenteelle. JCS määrittelee säännöt, joiden avulla JSON sarjallistetaan, ja sen avulla voidaan taata, että JSON on standardoitu ennen tiivisteen tai allekirjoituksen luomista. JSON-kanonikalisoinnille ei ole olemassa yhtä hyväksyttyä kanosointistandardia. Tässä dokumentissa viitataan IETF JSON Canoncalization Scheme-standardiin.

JSON kanonikalisointivaatimuksista voi lukea lisää dokumentista RFC 8785 JSON Canonicalization Scheme (JCS) [8]. Allekirjoitettavien JSON-muotoisten FHIR resurssien kanonisen muodon luomiseen täytyy käyttää IETF JSON Canonicalization Scheme (JCS) [9] IETF RFC 8785 mukaista toteutusta.

JCS [8] (Luku 3.2) kanonikalisointisäännöstö:

1. JSON-elementtien väliin ei saa sisällyttää välilyöntejä.

2. Primitiivisten datatyyppien sarjallistus: Joidenkin primitiivisten datatyyppien arvot voivat muuttua, jos ne sarjallistetaan ES6-standardin mukaisesti.

- Literaalit: JSON literaalit "null", "true" ja "false" eivät muutu sarjallistamisen johdosta.

- Merkkijonot: Jos jokin merkki kuuluu ASCII-merkkialueeseen (U+0000 - U+001F), se on sarjallistettava käyttäen pientä heksadesimaalista Unicode-merkintää (\uhhhh). Tähän on poikkeuksena valmiiksi määritellyt JSON-ohjausmerkit U+0008, U+0009, U+000A, U+000C tai U+000D, jotka täytyy sarjallistaa vastaavasti \b, \t, \n \f ja \r. Jos merkki ei kuulu ASCII-merkkialueeseen, se täytyy serialisoida sellaisenaan. Tähän on poikkeuksena merkit U+005C tai U+0022, jotka täytyy sarjallistaa vastaavasti \\ ja \". Lisäksi, tuloksena olevan Unicode-koodipisteiden sekvenssin täytyy olla ympäröity kaksinkertaisilla lainausmerkeillä (").
- Kokonaisluvut: RFC-standardin mukaan kokonaisluvut täytyy serialisoida ECMA-262 spesifikaation kohdan 7.1.12.1 mukaisesti [12]. RFC-standardi suosittelee sarjallistuksen esimerkki-implementaationa RYU, jolla on tehty JCS open-source Java implementaatio.

3. Kanonikalisen JSONin vaatimuksiin kuuluu, että JSON-objektin ominaisuudet ovat aakkosjärjestyksessä. Aakkosjärjestykseen kuuluu seuraavat säännöt

- JSON-objektin ominaisuudet täytyy järjestää rekursiivisesti. Tämä tarkoittaa sitä, että myös JSON-lapsiobjektien ominaisuudet täytyy järjestää.
- JSON-taulukon elementit täytyy järjestää. Kuitenkin, taulukon elementtien järjestystä ei saa muuttaa. Kun JSON-objektin ominaisuuksia lajitellaan, täytyy noudattaa seuraavia sääntöjä:
    - Järjestysprosessi koskee merkkijonoja niiden "raakana" muotona. Esimerkiksi rivinvaihtomerkkiä \n kohdellaan merkkinä U+000A.
    - Järjestettävät JSON-objektit formatoidaan taulukkoina UTF-16 mukaan. UTF-16 arvot järjestetään suoraan kokonaislukuina riippumatta kieliasetuksista.
    - JSON-objektin ominaisuudet järjestetään nimien mukaan nousevaan aakkosjärjestykseen seuraavasti: "", "a", "aa", "ab"..

4. UTF-8 Koodaus: Jotta voidaan varmistua alustariippumattomasta esityksestä, JSON-data täytyy koodata UTF-8 muotoon.

Huom. RFC 8785 [8] standardissa kohdassa 3.2.3 on esitetty esimerkki, jonka mukaisesti kanonikalisoinnin tulee toimia.

Kanonikalisointi voidaan suorittaa ulkoisella kanonikalisointiohjelmistolla tai -kirjastolla:

1. Luo allekirjoitettava JSON.

2. Käytä kanonikalisointiohjelmistoa järjestämään ja muuntamaan data kanonikalisoituun muotoon.
3. Allekirjoita kanonikalisoitu JSON.
4. Lisää tuloksena oleva allekirjoitusarvon alkuperäiseen JSON-tiedostoon sille määritettyyn rakenteeseen.
5. Järjestä valmis JSON käyttäen työkalua.

Yhteensopiva allekirjoituksen varmistusjärjestelmä:

1. Käsittelee ja parsii allekirjoitetun JSON tiedoston käyttäen JSON työkalua.
2. Lukee annetusta rakenteesta allekirjoituksen.
3. Poistaa allekirjoituksen käsitellystä JSONista.
4. Järjestää jäljellä olevat JSONin tiedot käyttäen JSON-työkalua.
5. Antaa ulkopuolisen kanonisointiohjelmiston järjestää ja palauttaa tietoa.
6. Varmistaa, että kanonikalisoidut tiedot vastaavat tallennettua allekirjoitusta käyttämällä allekirjoituksen luomiseen käytettyä algoritmia ja yksilöityä avainta.

4.5 JWS otsikon (headerin) elementit

JWS, FHIR ja JAdES spesifikaatiot sisältävät vaatimuksia JWS-otsikon elementeille. FHIR pakollisia JWS-otsikon elementtejä ovat alg, typ ja x5c.

JAdES standardin [3] ETSI TS 119 182-1 V1.1.1 (2021-03) 5.8.2.3.1 mukaisesti pakollisia ovat myös sigT, sigD, b64 ja crit elementit.

JSON Web Token (JWT) spesifikaation [5] IETF RFC 7519, mukaisia elementtejä esim. iat, issuer, exp ja aud voi lisätä JWS headeriin, mutta ne eivät ole pakollisia JAdES ja FHIR spesifikaatioissa, joten niitä ei käytetä.

4.5.1 Alg

Alg (algoritmi) otsikkoparametri identifioi JWS:n suojaamiseen käytetty allekirjoitusalgoritmin. Kelan Kanta HL7 FHIR allekirjoituksissa allekirjoituksissa käytettävät algoritmit ovat kappaleessa 3.2.2 . Alg-otsikkoparametrilla on oltava IETF RFC 7515:n [5] kohdassa 4.1.1 määritetty muoto ja arvo. Tämä otsikkoparametri on pakollinen. FHIR bundlen allekirjoituksessa käytössä olevat algoritmit on määritelty luvussa 3.1.2.

4.5.2 Typ

Typ (tyyppi) otsikkoparametri määrittää JWS:n mediatyypin [IANA.MediaTypes]. Typ otsikkoparametrin arvolla "JOSE" ilmaistaan, että tämä objekti on JWS tai JWE, JWS Compact Serialization tai JWE Compact Serialization muodossa. Typ arvolla "JOSE+JSON" ilmaistaan,

että tämä objekti on JWS tai JWE JWS JSON tai JWE JSON muodossa. FHIR bundlen allekirjoituksessa Typ parametrin arvo on JOSE.

4.5.3 B64

b64 otsikkoparametri ilmaisee onko JWS hyötykuorma base64url enkoodattu tai enkoodamaton. b64 arvolla true hyötykuorman pitää olla base64url enkoodattu. b64-otsikkoparametrilla on oltava IETF RFC 7797 [14]:n lausekkeessa 3 määritelty muoto ja arvo. b4 täytyy olla true, jos mId-elementin arvo on "http://uri.etsi.org/19182/ObjectIdByURI". FHIR bundlen allekirjoituksessa b64 parametrin arvo on true.

4.5.4 Crit

crit - parametri on JSON-taulukko, jossa pitää olla kaikkien JAdES-allekirjoituksien otsikkoparametrien nimet, kuten alg, b64, x5c, sigT ja sigD. Crit-otsikkoparametrilla on oltava IETF RFC 7797 [14]:n lausekkeessa 4.1.11 määritelty muoto ja merkitys.

4.5.5 x5c

x5c (X.509 Certificate Chain) otsikkoparametri on JSON-taulukko, jonka ensimmäisen elementin tulee sisältää allekirjoitukseen käytetyn sertifikaatin julkinen avain. Jokainen taulukon merkkijono on base64-koodattu ([RFC4648]:n osio 4 -- ei base64url-koodattu) DER [ITU.X690.2008] PKIX-sertifikaatin arvo.

4.5.6 sigT

sigT otsikkoparametri on merkkijono, joka sisältää UTC-muotoisen aikaleiman, jolloin allekirjoitus on tehty. Esim. "2019-11-19T17:28:15+03:00". Aikaleiman pitää olla muotoiltu IETF RFC 3339:n mukaisesti [12]. Aikaleima ei saa sisältää sekuntien murto-osaa vastaavaa osaa.

4.5.7 sigD

sigD otsikkoparametri on JSON-objekti, joka sisältää elementtejä, joilla viitataan yhteen tai useampaan hyötykuorma-tieto-objektiin. SigD määrittää miten viittaukset hyötykuorma-objekteihin käsitellään ja määrittää mekanismin edellä mainitujen vaatimuksen täyttämiseksi. sigD otsikkoparametri on JSON-objekti, joka sisältää elementtejä, joilla viitataan yhteen tai useampaan hyötykuorma-tieto-objektiin. SigD määrittää miten viittaukset hyötykuorma-objekteihin käsitellään ja määrittää mekanismin edellä mainitujen vaatimuksen täyttämiseksi. SigD-elementti Mechanism Id (mId) arvolla 'http://uri.etsi.org/19182/ObjectIdByURI' pitää sisältää mId ja ctys jäsenet. Mekanismia 'http://uri.etsi.org/19182/ObjectIdByURI' käytettäessä viittauksia ja tiivisteitä (pars, hashM ja hashV) ei lisätä sigD-elementtiin.

- mId - URI, joka identifioi käytettävän mekanismin, jota käytetään hyötykuorma-tietoobjekteihin viitattaessa ja niiden käsittelyssä hyötykuorman rakentamiseen.

- ctys - JSON-taulukko eli lista, joka sisältää jokaisen hyötykuorma-tietoobjektien sisältötyyppin, joihin viitataan samassa pars-taulukon kohdassa. FHIR bundlen allekirjoituksessa viittauksia ei käytetä, jolloin listassa ilmoitetaan ainoastaan hyötykuorman sisältötyyppi.

4.5.8 srCms

srCms otsikkoparametri on JSON-objekti, joka sisältää elementtejä, jotka määrittää allekirjoittajan allekirjoituksen yhteydessä tekemän sitoumuksen. Sitoumus voi olla määritelty osaksi allekirjoituspolitiikkaa tai rekisteröity tyyppi. Ennalta määritettyjen sitoumustyyppitunnisteiden luettelo on ETSI TS 119 172-1 [i.7] määrittelyssä. srCms arvon tulee olla sama kuin FHIR Signature -elementin type-parametrissä. FHIR spesifikaation [1] FHIR Specification (v4.0.1: R4 - STU) mukaan allekirjoituksen täytyy sisältää srCms-elementti.

- srCms on JSON-taulukko, joka sisältää JSON-objekteja, joilla on seuraavia elementtejä:
- commId - OId-tyyppinen tieto, jonka arvona täytyy olla URI, joka yksilöi yhden allekirjoittajan tekemän sitoumuksen.
- commQuals - Taulukko sitoumusta koskevista lisätiedoista (objekteja).

5 Allekirjoituksessa käytetyt prosessit (normatiivinen)

5.1 Allekirjoituksen muodostaminen

Yhden FHIR bundlen allekirjoituksen muodostaminen (ObjectIdByURI menetelmä)

1. Valmistele JWS otsikko
2. Poista Signature-elementti allekirjoitettavasta bundlesta
3. Kanonikalisoi allekirjoitettava bundle vaatimusten mukaisella JSON kanonikalisointiohjelmalla
4. Base64 enkoodaa kanonikalisoitu bundle (hyötykuorma) vaatimusten mukaisella base64UrlEncode-ohjelmalla
5. Kanonikalisoi JWS otsikko vaatimusten mukaisella JSON kanonikalisointiohjelmalla
6. Luo allekirjoitus yksityisellä avaimella ja vaatimusten mukaisella allekirjoitusalgoritmilla
7. Luo irroitettu (detached) allekirjoitus poistamalla hyötykuorma JWS:stä
8. Lisää irroitettu (detached) allekirjoitus alkuperäisen bundlen Signature-elementtiin base64 enkoodattuna

5.2 Allekirjoituksen tarkastaminen

Allekirjoitus tarkistetaan seuraavasti:

Bundlen allekirjoituksen tarkistus (ObjectIdByURI menetelmä)

1. Pura JWS allekirjoitus tarkistettavan resurssin Signature-elementistä
2. Poista Signature-elementti tarkistettavasta resurssista
3. Kanonikalisoi tarkistettava nykyinen bundle vaatimusten mukaisella JSON kanonikalisointiohjelmalla (IETF JSON Canonicalization Scheme (JCS))
4. Base64 enkoodaa kanonikalisoitu bundle vaatimusten mukaisella base64UrlEncode-ohjelmalla
5. Muodosta tarkistettava JWS alkuperäisestä JWS-otsikosta, nykyisestä resurssista ja alkuperäisestä JWS:n allekirjoitusosasta
6. Pura tarkistukseen käytettävä julkinen avain JWS-otsikon x5c elementistä
7. Tarkista referenssi JWS:n allekirjoitus saadulla julkisella avaimella

Allekirjoituksen muut tarkistukset:

- Allekirjoitusalgoritmi (alg-elementti) on sallittu (3.2.2. Allekirjoitusalgoritmit ja menetelmät)
- typ parametrin arvo on ”jose” tai ”jose+json”
- crit-elementin parametrilistassa mainitut parametrit (alg, b64, x5c, sigT, sigD ja srCms) ovat JWS-otsikossa ja oikein
- x5c parametrin sertifikaatti on pätevä ja sertifikaatin myöntäjä (Issuer) on luotettu
- sigD rakenne on oikein ja sisältää vaaditut parametrit vaaditussa muodossa
- Allekirjoitussertifikaatin (x5c-elementti) 'Validity Not Before' on vanhempi kuin allekirjoituksen aikaleima (sigT-elementti)
- Allekirjoitussertifikaatti (x5c-elementti) 'Validity Not After' on myöhempi kuin allekirjoituksen aikaleima (sigT-elementti)
- Allekirjoitussertifikaattia ei ole revokoitu

6 Esimerkit (ei normatiivisia)

6.1 Yhden Bundle-resurssin allekirjoitus

6.1.1 JWS-allekirjoitus

```
// Header
{
  "alg": "RS256",
  "typ": "JOSE",
  "b64": true,
  "crit": [
    "b64",
    "alg",
    "typ",
    "x5c",
    "sigT",
    "sigD",
    "srCms"
  ],
  "x5c": [
    "MIIEtDCCApygAwIBAgIFHyPzaocwDQYJKoZIhvcNAQELBQAwgZsxCzAJBgNVBAYTAkl
FMRAwDgYDVQQIDAdLSUxEQVJFMGvnaV2c2UhFqIAo156r41Dnjng3qiZNc="
  ],
  "x5t": "6e052fc5fbfea7f6cd0fbc4061c798f6f55e8608",
  "sigT": "2023-07-05T12:48:48Z",
  "sigD": {
    "mId": "http://uri.etsi.org/19182/ObjectIdByURI",
    "ctys": [
      "text/json"
    ]
  },
  "srCms": [
    {
      "commId": "1.2.840.10065.1.12.1.13",
      "commQuals": [
       {"system":"urn:iso-astm:E1762-95:2013","display":"Review Signature"}
      ]
    }
  ]
}.
{
    "resourceType": "Bundle",
    "id": "ab71e831-3e8e-40c5-b005-cbbbf6f30423",
    "type": "collection",
    "entry": [
       ...
    ]
}


// Serialized Detached Signature
          BASE64URL(UTF8(JWS Header) || '..' || BASE64URL(JWS signature)
```

6.1.2 Signature-elementti

```
"signature" : {[
    "type" : {
      "system" : "urn:iso-astm:E1762-95:2013",
      "code" : "1.2.840.10065.1.12.1.13",
      "display" : "Review Signature "
    }],
    "when" : "2022-02-08T10:16:32.000+10:00",
    "who": {
      "identifier" : {
           "system" : "urn:ietf:rfc:3986",
           "value" : "urn:oid:1.2.246.xxxx.yyyy.zzz"
      },
      "display" : "Organisaation nimi"
    },
    "targetFormat" : "application/fhir+json",
    "sigFormat" : "application/jose",
    "data" : "UEQ5NGJXd2dkbVZ5YzJsdmJqMGlNUzR3SWlCbGJtTnZaR2x1WnowaVZWUkdMVGdp
Li4xY21VK0Nqd3ZSVzUyWld4dmNHVStJQT09"
}
```