Vergelijking REST API Signing Standaarden

KPAPI Handreiking
Werkversie 30 september 2022

Deze versie:
https://phaasnoot.github.io/REST_API_Signing_Standaarden/

Laatst gepubliceerde versie:
https://publicatie.centrumvoorstandaarden.nl/dk/template/

Laatste werkversie:
https://phaasnoot.github.io/REST_API_Signing_Standaarden/

Redacteurs:
Carmen Visinescu (Kadaster)
Peter Haasnoot (Logius)

Auteurs:
Carmen Visinescu (Kadaster)
Peter Haasnoot (Logius)

Doe mee:
GitHub PHaasnoot/REST_API_Signing_Standaarden
Dien een melding in
Revisiehistorie
Pull requests

Samenvatting

In dit document worden de volgende standaarden beschreven en met elkaar vergeleken:

- JAdES [JAdES]
- HTTP Message Signatures (httpbis-message-signatures) [HTTP-MessageSig]

Status van dit document

Dit is een werkversie die op elk moment kan worden gewijzigd, verwijderd of vervangen door andere documenten. Het is geen door de werkgroep goedgekeurde consultatieversie.

Inhoudsopgave

1. Vergelijking JAdES vs httpbis-message-signatures

Dit onderdeel is niet normatief.

1.1 Inleiding

In deze handreiking wordt eerst ingegaan op de algemene aspecten van signing, daarna worden de signing standaarden JAdES en http-message-signatures in hoofdlijnen beschreven en wordt aangeven wat de voornaamste eigenschappen en functionele aspecten zijn van beide standaarden.

1.2 Digital Signature algemeen

Digitale handtekeningen zijn ontworpen om de integriteit van de inhoud en authenticatie van de oorsprong en onweerlegbaarheid van de oorsprong te bieden aan gegevensobjecten die door netwerken van hun oorsprong naar hun bestemming worden getransporteerd. Digitale handtekeningen zijn gebaseerd op asymmetrische (of openbare sleutel) cryptografie. Het digitaal ondertekenen van een gegevensobject bestaat uit twee stappen :

1. het berekenen van de samenvatting van de binaire representatie van het te ondertekenen data-object – de digest-data
2. het resultaat van deze operatie versleutelen (encrypten) met de privésleutel van de ondertekenaar. (ETSI (European Standards Institute TR 119 001) gebruikt het term "digital signature value" voor het resultaat van de laatste stap)

De proces-verificatie bevat de volgende stappen:

1. Voorbereiden van de "to-be-signed-data" met de canonicalisatie- geaccepteerde transformaties en het berekenen van het digest-data (bv. met behulp van een hash algorithme)
2. Decrypt de ondertekende digital signature waarde met de public-key
3. Vergelijk beide waarden van het oorspronkelijke bericht met het resultaat van stap 2. Als de waarden gelijk zijn, dan is de verificatie succesvol.

De digitale ondertekening maakt gebruik van een Public Key Infrastructure (welke verder gebruik maakt van entiteiten zoals certificaten, CRLs, OCSP antwoorden, time-stamp tokens (die bewijzen dat een bepaald data-object bestond op een bepaald moment), Trust services, zoals services uitgevers certificaten (CAs), services uitgevers time-stamp tokens, etc.

Alle standaard formaten bevatten de volgende componenten:

- het "signed data object"
- het "digital signature" veld

- het "signing certificaat" (of het certificaat pad)
- de objecten genoemd ‘signed attributen’ optionele of verplichte velden
- de objecten genoemd : ‘unsigned attributen’ toegevoegd op een gegeven moment ook, nadat de "digital signature" is gerealiseerd

Een ‘signed’ attribuut is een object, toegevoegd in de "digitaal signature" structuur. Het is een binair deel van het berekenen van het digitale signature veld.

Afhankelijk van het business/juridische-scenario worden bepaalde velden in het ’signed-attributen’ en/of unsigned attributen structuur wel/niet toegevoegd.

1.3 Http Message Signatures

Op moment van schrijven is dit de laatste versie van de standaard: draft-ietf-httpbis-message-signatures-11 [HTTP-MessageSig].

HTTP Message Signatures ondersteunt gebruiksscenario's waar het volledige HTTP-bericht mogelijk niet bekend is bij de ondertekenaar, en waar het bericht onderweg kan worden getransformeerd (bijvoorbeeld door tussenpersonen) voordat deze de ontvanger bereikt. De standaard beschrijft ook een middel voor de verzender van een ‘request’ om te verzoeken om een handtekening van de ontvanger toe te passen op de ‘response’ in een lopende HTTP-uitwisseling.

De standaard geeft specifiek invulling aan:

- Een algemene nomenclatuur en canonieke regels voor de verschillende protocolelementen en andere componenten van HTTP-berichten die gebruikt worden om de handtekeningbasis te maken.
- Algoritmen voor het genereren en verifiëren van handtekeningen via HTTP berichtcomponenten die deze handtekeningbasis gebruiken door toepassing van cryptografische primitieven.
- Een mechanisme voor het toevoegen van een handtekening en gerelateerde metadata aan een HTTP-bericht en voor het ontleden van bijgevoegde handtekeningen en metagegeven van HTTP-berichten. Om dit te vergemakkelijken, definieert de standaard de "Signing-Input" en "Signature" velden.

Specifieke features van de standaard:

- Transformation resistant: Er wordt rekening mee gehouden dat de HTTP standaard toestaat dat de volgorde van HTTP elementen in een bericht kan veranderen.
- Canonicalization: Regels geven aan hoe elementen naar een standaard representatie worden vertaald (geschikt voor ondertekening)
- Components: Er worden componenten van een HTTP bericht gedefinieerd om fijnmazig onderdelen te kunnen benoemen/ondertekenen (bv: query, uri etc)
- Multiple Signatures: Er is ondersteuning voor meerdere signatures, bv ondertekening door een proxy onderweg;
- Requesting Signatures: Een partij in de communicatie kan expliciet aangeven dat hij/zij een ondertekend bericht verwacht van de andere partij;
- Request-Response Signature Binding: Een response kan aan de signature van de request refereren om expliciet aan te geven (en aan te tonen) dat de response antwoord is op de specifieke request;
- TLS-Terminating Proxies: Een specifieke use-case voor de standaard is het omzetten van een via tweezijdig TLS ontvangen bericht naar het achterland van een API gateway; het ontvangen certificaat kan meegezonden worden voor verdere afhandeling, de gateway kan het zelf ondertekenen met het eigen

certificaat om aan te geven dat de gateway het bericht (en certificaat) correct heeft ontvangen (en gevalideerd);

1.4 JAdES

ETSI heeft een standaard voor JWS ontwikkeld, naast de andere ETSI standaarden voor de AdES digital signatures van het Regulation (EU) No 910/2014, genoemd JAdES (ETSI TS 119 182-1). JAdES ondersteunt veilige communicatie die voldoet aan de vereisten van de eIDAS-verordening van de Europese Unie (nr. 910/2014) voor geavanceerde elektronische handtekeningen en zegels en wettelijke vereisten voor diensten zoals open bankieren.

“In samenwerking met Open Banking Europe heeft ETSI een oplossing ontwikkeld die voldoet aan de vereisten van Open Banking API's en tegelijkertijd de authenticiteit van financiële transacties garandeert.”

TS 119 182-1 is op RFC 7515 gebaseerd. Er zijn nieuwe attributen, header-parameters toegevoegd en er worden ook 4 "baseline levels" geïntroduceerd.

Het huidige profile (JSON signature) is gelijk met het basic JAdES-B-B level. Het maakt gebruik van JWS header parameters, zoals ook gedefinieerd in het JAdES profile.

ETSI TS 119 182-1 kan worden gebruikt voor elke transactie tussen een persoon en een bedrijf, tussen twee bedrijven, tussen een persoon en een overheidsinstantie, enz. die van toepassing is op elektronische communicatie. De technische kenmerken van de specificatie kunnen daarom worden toegepast op het gebruik van op PKI gebaseerde digitale handtekeningtechnologie.

Ontwerp principes:

- De handtekeningen die voldoen aan dit profiel ondersteunen het gebruik van gekwalificeerde certificaten voor elektronische zegels in overeenstemming met Commission Delegated Regulation (EU) 2018/389
- Het profiel is afgestemd op JAdES baseline digitale handtekeningen zoals gespecificeerd door ETSI
- De handtekening beschermt een HTTP-body en optioneel geselecteerde HTTP-headervelden.
- Een enkele handtekening moet worden opgenomen in een HTTP-header die is losgekoppeld (detached) van het payload-message
- Het profiel is bedoeld om de interoperabiliteit te maximaliseren
- Er worden geen beperkingen opgelegd aan de inhoud van de ondertekende payloads. Het kan worden gebruikt om JSON, XML ISO 20022 of elke andere vorm van gegevens te beschermen
- JSON Web Signature-headers en HTTP-headervelden die essentieel zijn voor de beveiliging van de uitwisseling, evenals HTTP Body, zijn zodanig beveiligd dat ze niet gewijzigd kunnen worden .
- JAdES-handtekeningen kunnen geserialiseerd worden met behulp van: JWS Compact Serialization of JWS JSON Serialization zoals gespecificeerd in IETF RFC 7515 [2].
- Handtekeningen kunnen later worden gebruikt als bewijs in de rechtbank (d.w.z. zijn "onweerlegbaar").
- JAdES-handtekeningen met de profielen B-LT, B-LTA kunnen de digitale certificaat validiteit op de lange termijn garanderen. Deze profielen zijn geschikt voor de realisatie van de archiveringseisen.
- Handtekeningen kunnen zowel op HTTP-requests en ook op HTTP-responses worden toegepast.

1.5 Vergelijking en Use Cases

1.5.1 HTTP Message Signatures (Httpbis-message-signatures) use-cases:

- TLS-Terminating proxy use-case: De tweezijdig TLS ontvangen berichten via een API gateway kunnen middels de httpbis-message signature ondertekend worden met het eigen gateway certificaat. Dat is om aan te geven dat de gateway (en certificaat) het bericht correct , zonder transformaties heeft ontvangen (en gevalideerd); End-to-end communicatiebeveiliging en berichtenintegriteit voor de berichten http-message-componenten en de http-derived-componenten tussen 2 systemen wordt in dit geval gerealiseerd.
- Realisatie van een additionele beveiligslaag in de uitwisselberichten tussen 2 systemen naast het transport tweezijdig TLS beveiligingscommunicatie.
- Httpbis-message-signature is gespecialiseerd in de ondertekening van een deel van de http-headers-velden en http-derived components
- Httpbis-message-signature introduceert een mechanisme van ’requesting-signature’ in de communicatie tussen 2 systemen
- Httpbis-message-signature introduceert een mechanisme van ’request-response signature binding’

1.5.2 JAdES use-cases:

- De ondertekende berichten (payload) worden juridisch gebruikt als bewijs in de rechtbank (d.w.z. zijn "onweerlegbaar").

JAdES-handtekening ondersteunt de veilige communicatie die voldoet aan de vereisten van de eIDAS-verordening van de Europese Unie (nr. 910/2014) voor geavanceerde of gekwalificeerd elektronische handtekeningen en zegels en wettelijke vereisten. In dit scenario kunnen geavanceerde /gekwalificeerd ondertekening certificaten van de PKI-QTPS’s of QTSA’s (gekwalificeerd timestamps certificaten) gebruikt worden.

JAdES-handtekening wordt gebruikt als standaard in de communicatie van de payment-services conform EU Payment Services Directive 2015/2366 [16] (PSD2).

- JAdES–handtekening kan gebruikt worden in de realisatie van een B2B betrouwbare veilige communicatie tussen 2 systemen op basis van de REST-API.

De ondertekening van de berichten en de payload-berichten kan makkelijk gerealiseerd worden met behulp van JAdES-B-B als een JAdES-compliant JWT detached. Het is een eenvoudige JSON-structuur ondertekening. Daardoor kan het makkelijk door de client-applicaties gebruikt en gerealiseerd worden.

JAdES–handtekening detached verandert ook de payload-structuur niet. Dat betekent dat een client die de validatie van handtekening niet ondersteunt, kan blijven werken alsof er geen handtekening is toegepast. JAdES–handtekening detached kan getransporteerd worden met behulp van een HTTP-header, waardoor de aanwezigheid niet opdringerig is en gemakkelijk is te transporteren.

In een juridisch context wordt de JAdES-handtekening verrijkt met verschillende velden in de handtekening bv: tijdstip ondertekening; commitment (reden) van de ondertekening; toevoegen van de counter-ondertekening;

- Realisatie van de archiveringseisen van de ondertekende communicatie berichten tussen 2 systemen.

Met de implementatie van de JAdES-B-LT en JAdES-B-LTA profielen kunnen de JAdES-handtekeningen oneindig worden verlengd t.b.v archiveringseisen

1.6 BIJLAGE A : HTTPBIS-Message-Signatures

Zie [HTTP-MessageSig]

Voorbeelden:

This section provides non-normative examples that may be used as test cases to validate implementation correctness. These examples are based on the following HTTP messages:

For requests, this test-request message is used:

```
NOTE: '\' line wrapping per RFC 8792

POST /foo?param=Value&Pet=dog HTTP/1.1
Host: example.com
Date: Tue, 20 Apr 2021 02:07:55 GMT
Content-Type: application/json
Content-Digest: sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+T\
  aPm+AbwAgBWnrIiYllu7BNNyealdVLvRwEmTHWXvJwew==:
Content-Length: 18

{"hello": "world"}
```

For responses, this test-response message is used:

```
NOTE: '\' line wrapping per RFC 8792

HTTP/1.1 200 OK
Date: Tue, 20 Apr 2021 02:07:56 GMT
Content-Type: application/json
Content-Digest: sha-512=:JlEy2bfUz7WrWIjc1qV6KVLpdr/7L5/L4h7Sxvh6sN\
  HpDQWDCL+GauFQWcZBvVDhiyOnAQsxzZFYwi0wDH+1pw==:
Content-Length: 23

{"message": "good dog"}
```

1.6.1 Signing components

This example covers additional components (the authority, the Content-Digest header field, and a single named query parameter) in test-request using the rsa-pss-sha512 algorithm.

The corresponding signature base is:

```
NOTE: '\' line wrapping per RFC 8792

"@authority": example.com
"content-digest": sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX\
  +TaPm+AbwAgBWnrIiYllu7BNNyealdVLvRwEmTHWXvJwew==:
"@query-param";name="Pet": dog
"@signature-params": ("@authority" "content-digest" \
  "@query-param";name="Pet")\
  ;created=1618884473;keyid="test-key-rsa-pss"
```

This results in the following Signature-Input and Signature headers being added to the message under the label sig-b22:

```
NOTE: '\' line wrapping per RFC 8792

Signature-Input: sig-b22=("@authority" "content-digest" \
  "@query-param";name="Pet");created=1618884473\
  ;keyid="test-key-rsa-pss"
Signature: sig-b22=:W2kxR52X0tXN9u7yjyPeWa0T3D0SVG8KkPo+lOWyb2TGdLz\
  ixWjUlbehjnNhzA+wFWnE6+hdKH8KR6Z9FvsxCc+44XrqxzT7Vcsror5SjMyfx6Nq\
  tELklj1u2L4JovANI80BSoVobSoc+v9NRVWJZU7WAVow8H2CucCcv2cy1tKFCTMyc\
  m9LQrIz63Tg5tcGWj64b12nmwj9TwwkCygfz0MTyIytjYLVzKw7mXpL4jGFZ5lsw2\
  VT2eB3qpF2d/Psy0p1heKhrkz9uvKeCoj+P5QjLMS4eirHDqpKqe9YmCaMsJAUYSU\
  M86qC8qO6vMQhTMegTkEe25DquVcTiOAEAw==:
```

1.7 BIJLAGE B : JAdES

1. JAdES-handtekening velden

Conform RFC 7515 en RFC 7797 (JSON Web Signature(JWS)) de elementen van de JSON ‘’digital signatures’’ zijn:

- JOSE Header (JSON Object Signing and Encryption):
- JWS Protected Header bevat de ’signed attributes’’ in het container-element ‘’etsiSigProps’’
- JWS Unprotected header bevat optioneel het container-header-element genoemd ‘’etsiU’’. Deze header parameter wordt in het JAdES Long Term Validation profile gebruikt.
- JWS Payload (’Data to be Signed’)
- JWS Signature (‘’Detached’’ van ‘’Data to be Signed’’)

2. JAdES Serialisations methoden

In alle serializatie-methoden zijn de JWS Protected Header, JWS Payload, and JWS Signature base64url encoded.

- JSON Compact Serialization Het kan alleen de ’protected headers’ bevatten.
  Het wordt alleen in het JAdES-BASELINE-B level gebruikt. Het JSON-container-element signed attributes ‘’etsiSigProps’’, is een deel van de JWS Protected Header.
- JSON Serialization Het kan beide type headers (protected en unprotected) bevatten en ondersteunt meerdere ondertekeningen van hetzelfde object. De JAdES-BASELINE-T/-LT/-LTA handtekening profielen kunnen dit serialization format gebruiken. Het JAdES- handtekening serialization format ondersteunt de realisatie van de ’multiple-signatures’ en ’counter-signatures’
- JSON Flattened Serialization Het kan gebruikt worden zoals één signature gebruikt wordt. De JAdES-BASELINE-T/-LT/-LTA signature profielen kunnen dit serialization format gebruiken. Deze JAdES-handtekening flattened serialization format ondersteunt de realisatie van de ’counter-signatures’. Het JWS object wordt als een HTTP header-veld: "x-jws-signature“ toegevoegd in de HTTP-bericht-headers.

3. JAdES-handtekening format

- ENVELOPING: het JWS Payload is enveloped in de JAdES-handtekening. Deze format kan alleen één handtekening bevatten. JAdES enveloping ondersteunt counter-signatures.

- DETACHED: het JWS Payload is niet toegevoegd in het JAdES signature.
  Om deze signature te valideren is het oorspronkelijke bericht nodig. JAdES-detached ondersteunt multiple-siganture (parallel-sigantures) en ook counter-signatures. Met JAdES-handtekening is het niet mogelijk alleen een deel van het JSON payload bericht te ondertekenen. Met JAdES-handtekening is het JSON

payload bericht volledig ondertekend. Indien nodig zouden additioneel de rest van de payload als aparte berichten gestuurd moeten worden naar de betreffende partij.

4. JAdES-handtekening profielen

- Het B-B-level bevat in de handtekening headerparameters en sommige niet-ondertekende componenten in de etsiU unsigned header-parameter.
- Het B-T-level bevat in de handtekening de elementen van het profiel B-B samen met een vertrouwd timestamp-token dat bewijst dat de handtekening zelf op een bepaald tijdstip daadwerkelijk bestond.
- Het B-LT-level bevat in de handtekening de elementen van het profiel B-T plus het materiaal dat nodig is voor het valideren van de handtekening in het handtekeningdocument. Dit niveau is bedoeld om de beschikbaarheid van het validatiemateriaal op lange termijn te realiseren.
- Het B-LTA-level bevat in de handtekening de elementen van het profiel B-LT plus de elektronische tijdstempels die de validatie van de handtekening lang na het genereren ervan mogelijk maken. Dit niveau is bedoeld om de beschikbaarheid en integriteit van het validatiemateriaal op lange termijn te realiseren.

A. Referenties

A.1 Informatieve referenties

[HTTP-MessageSig]
HTTP Message Signatures. URL: https://www.ietf.org/archive/id/draft-ietf-httpbis-message-signatures-11.html

[JAdES]
JAdES digital signatures. URL: https://www.etsi.org/deliver/etsi_ts/119100_119199/11918201/01.01.01_60/ts_11918201v010101p.pdf

↑