# Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni – Pattern di sicurezza

**Versione 1.1 del 19/05/2023**

**AGID | via Liszt, 21 – 00144 Roma | protocollo@pec.agid.gov.it | www.agid.gov.it | tel. 06 85264.1**

| Versione | Data | Tipologia modifica |
|---|---|---|
| 1 | 27/04/2021 | Prima emissione |
| **1.1** | 19/05/2023 | - Modificato titolo<br>- Modificata intestazione delle pagine<br>- Nel paragrafo 3.2 aggiunti il secondo e il terzo capoverso<br>- Nel paragrafo 4.3.1 aggiornato il secondo capoverso ed aggiunti i capoversi terzo, quarto e quinto<br>- Nel paragrafo 4.4.1 aggiornato il secondo capoverso ed aggiunti i capoversi terzo, quarto e quinto<br>- Aggiunto paragrafo 5.3<br>- Aggiunto capitolo 6 |

# Sommario

**Introduzione** ..... **5**
**Capitolo 1 Ambito di applicazione** ..... **8**
1.1 Soggetti destinatari ..... 8
**Capitolo 2 Riferimenti e sigle** ..... **9**
2.1 Note di lettura del documento ..... 9
2.2 Standard di riferimento ..... 9
2.3 Termini e definizioni ..... 10
**Capitolo 3 Sicurezza di canale e/o identificazione delle organizzazioni** ..... **11**
3.1 [ID_AUTH_CHANNEL_01] Direct Trust Transport-Level Security ..... 11
3.1.1 Descrizione ..... 11
3.1.2 Regole di processamento ..... 12
3.2 [ID_AUTH_CHANNEL_02] Direct Trust mutual Transport-Level Security ..... 13
3.2.1 Descrizione ..... 13
3.2.2 Regole di processamento ..... 14
**Capitolo 4 Accesso del soggetto fruitore** ..... **16**
4.1 [ID_AUTH_SOAP_01] Direct Trust con certificato X.509 su SOAP ..... 16
4.1.1 Descrizione ..... 16
4.1.2 Regole di processamento ..... 17
4.1.3 Esempio ..... 18
4.2 [ID_AUTH_SOAP_02] Direct Trust con certificato X.509 su SOAP con unicità del token/messaggio ..... 20
4.2.1 Descrizione ..... 21
4.2.2 Regole di processamento ..... 21
4.2.3 Esempio ..... 23
4.3 [ID_AUTH_REST_01] Direct Trust con certificato X.509 su REST ..... 26
4.3.1 Descrizione ..... 26
4.3.2 Regole di processamento ..... 27
4.3.3 Esempio ..... 29
4.4 [ID_AUTH_REST_02] Direct Trust con certificato X.509 su REST con unicità del token/messaggio ..... 30
4.4.1 Descrizione ..... 30
4.4.2 Regole di processamento ..... 32
4.4.3 Esempio ..... 33

**Capitolo 5 Integrità ..... 36**
5.1 [INTEGRITY_SOAP_01] Integrità del payload del messaggio SOAP ..... 36
5.1.1 Descrizione ..... 36
5.1.2 Regole di processamento ..... 37
5.1.3 Esempio ..... 38
5.2 [INTEGRITY_REST_01] Integrità del payload messaggio REST ..... 40
5.2.1 Descrizione ..... 40
5.2.2 Regole di processamento ..... 41
5.2.3 Esempio ..... 43
5.3 [INTEGRITY_REST_02] Integrità del payload delle request REST in PDND 44
5.3.1 Descrizione ..... 45
5.3.2 Regole di processamento ..... 46
5.3.3 Esempio ..... 48
**Capitolo 6 Inoltro dati tracciati nel dominio del Fruitore ..... 50**
6.1 [AUDIT_REST_01] Inoltro dati tracciati nel dominio del Fruitore REST ..... 50
6.2 [AUDIT_REST_02] Inoltro dati tracciati nel dominio del Fruitore REST con correlazione 58

# Introduzione

I pattern di sicurezza definiscono le modalità per assicurare che le interazioni tra fruitore ed erogatore siano realizzate nel rispetto delle specifiche esigenze di sicurezza determinate dalla natura delle transazioni realizzate e dalle prescrizioni normative che riguardano le stesse.

I pattern di sicurezza si applicano ai pattern di interazione indicati nel Documento Operativo - Pattern di interazione, e sono scelti dall'erogatore in funzione alle specifiche esigenze applicative in relazione alla natura dei fruitori.

Il Documento operativo descrive i pattern di sicurezza individuati da AgID che gli erogatori DEVONO utilizzare per soddisfare le necessità individuate dai requisiti funzionali e non funzionali delle specifiche interazioni con i propri fruitori.

Data la variabilità nel tempo delle esigenze delle amministrazioni e delle tecnologie abilitanti, nonché considerata la natura incrementale del ModI, l'elenco dei pattern di sicurezza non è da intendersi esaustivo. Nel caso in cui un'amministrazione abbia esigenze non ricoperte nei seguenti pattern di sicurezza DEVE informare AgID, nei modi indicati nel capitolo 7 «Pattern e profili di interoperabilità» delle Linee di indirizzo sull'interoperabilità tecnica delle Pubbliche Amministrazioni. Le tecnologie e standard per assicurare la sicurezza dell'interoperabilità tramite API utilizzabili nel ModI, tra cui OAuth 2.0, sono individuate nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

I pattern di sicurezza individuati coprono gli aspetti di comunicazione «sicura» tra i domini delle singole parti. Le parti mantengono la loro autonomia negli aspetti organizzativi e di sicurezza interni al proprio dominio.

I pattern di sicurezza:

- definiscono a livello di specifica tecnologica uno «strumento condiviso» utile a favorire l'interoperabilità tra erogatori e fruitori.
- forniscono un comune linguaggio per fruitori ed erogatori utile a trattare le necessità e le caratteristiche delle interfacce di servizio.
- offrono agli sviluppatori le modalità tecniche supportate da standard tecnologici documentati, revisionati e testati per esporre i servizi digitali.

I pattern di sicurezza affrontano il tema della sicurezza su due livelli differenti:

- Canale: definisce le modalità di trasporto dei messaggi tra i confini dei domini delle entità coinvolte.
- Messaggio: definisce le modalità di comunicazione dei messaggi tra componenti interne dei domini delle entità coinvolte.

Ogni pattern di sicurezza è strutturato come segue:

- Descrizione: rappresentazione in linguaggio naturale del profilo con relativi precondizioni e obiettivi.
- Regole di processamento: elenco dei passi da eseguire per implementare il profilo.
- Tracciato: ove presente, fornisce un esempio dei messaggi prodotti nell'interazione.

Gli erogatori, ove necessario in accordo con i fruitori, a seguito dell'analisi dei requisiti realizzata, per individuare le proprie esigenze funzionali e non funzionali, DOVREBBERO:

- individuare tra i pattern di interazione (vedi Documento operativo - Pattern di interazione) quelli che soddisfano le proprie esigenze;
- individuare tra i pattern di sicurezza quelli che soddisfano le proprie esigenze;
- implementare le interfacce di servizio attraverso la combinazione dei pattern di interazione e di pattern di sicurezza.

L'individuazione dei pattern di sicurezza DEVE ricoprire solamente i requisiti necessari.

Il Trust è uno dei mezzi più importanti per gestire le problematiche di sicurezza nello scambio di informazione in rete per consentire l'interoperabilità tra i sistemi. Esso si basa sul reciproco riconoscimento delle entità interagenti e sulla fiducia nei rispettivi comportamenti.

Nel presente Documento operativo, per direct trust si intende la relazione di fiducia tra fruitore ed erogatore, stabilita in modalità diretta, attraverso accordi che si basano sulla condivisione del reciproco modus operandi.

Si rimanda alle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale), in merito agli algoritmi utilizzabili per la corretta implementazione dei pattern di sicurezza.

Capitolo 1

# Ambito di applicazione

Il presente Documento operativo, allegato delle Linee Guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni, definisce i pattern di sicurezza definiti nel ModI.

## 1.1 Soggetti destinatari

Il Documento Operativo è destinato ai soggetti di cui all'articolo 2, comma 2 del CAD, così come indicato dall'articolo 75 dello stesso. I destinatari la attuano nella realizzazione dei propri sistemi informatici che fruiscono o erogano dati e/o servizi digitali ad altri soggetti.

Il Documento Operativo è rivolto ai soggetti privati che devono interoperare con la Pubblica Amministrazione per erogare o fruire di dati e servizi tramite sistemi informatici.

Capitolo 2

# Riferimenti e sigle

## 2.1 Note di lettura del documento

Conformemente alle norme ISO/IEC Directives, Part 3 per la stesura dei documenti tecnici la presente linea guida utilizzerà le parole chiave «DEVE», «DEVONO», «NON DEVE», «NON DEVONO», «E" RICHIESTO», «DOVREBBE», «NON DOVREBBE», «RACCOMANDATO», «NON RACCOMANDATO» «PUO"» e «OPZIONALE», la cui interpretazione è descritta di seguito.

- **DEVE** o **DEVONO**, indicano un requisito obbligatorio per rispettare la linea guida;
- **NON DEVE** o **NON DEVONO**, indicano un assoluto divieto delle specifiche;
- **DOVREBBE** o **RACCOMANDATO** o **NON DOVREBBE** o **NON RACCOMANDATO**, indicano che le implicazioni devono essere comprese e attentamente pesate prima di scegliere approcci alternativi;
- **PUÓ** o **POSSONO** o l'aggettivo **OPZIONALE**, indica che il lettore può scegliere di applicare o meno senza alcun tipo di implicazione la specifica.

## 2.2 Standard di riferimento

Sono riportati di seguito gli standard tecnici indispensabili per l'applicazione del presente documento.

| | |
|---|---|
| **[ISO 19115]** | UNI EN ISO 19115:2005, Informazioni geografiche – Metadati |
| **[RFC3230]** | Instance Digests in http |
| **[RFC3744]** | Web Distributed Authoring and Versioning (WebDAV) Access Control Protocol |
| **[RFC5246]** | The Transport Layer Security (TLS) Protocol Version 1.2 |
| **[RFC7231]** | Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content |

| | |
|---|---|
| **[RFC7233]** | Hypertext Transfer Protocol (HTTP/1.1): Range Requests |
| **[RFC7515]** | JSON Web Signature (JWS) |
| **[RFC7519]** | JSON Web Token (JWT) |
| **[RFC8725]** | JSON Web Token Best Current Practice |

## 2.3 Termini e definizioni

Di seguito si riportano gli ACRONIMI che verranno utilizzati nella presente Linee Guida:

| | |
|---|---|
| **[AgID]** | Agenzia per l'Italia Digitale |
| **[CAD]** | Codice Amministrazione Digitale, D.lgs. 7 marzo 2005, n. 82 |
| **[PA]** | Pubblica Amministrazione |
| **[UML]** | Linguaggio di modellazione unificato (Unified Modeling Language) |
| **[RPC]** | Remote procedure call |
| **[SOAP]** | Simple Object Access Protocol |
| **[REST]** | Representational State Transfer |

Capitolo 3

# Sicurezza di canale e/o identificazione delle organizzazioni

Di seguito le indicazioni per le tecnologie accolte dal ModI.

L'AgID assicura l'aggiornamento degli stessi per soddisfare le esigenze espresse dalle PA.

## 3.1 [ID_AUTH_CHANNEL_01] Direct Trust Transport-Level Security

Comunicazione tra fruitore ed erogatore che assicuri, a livello di canale:

- confidenzialità;
- integrità;
- identificazione dell'erogatore, quale organizzazione;
- difesa dalle minacce derivanti dagli attacchi: Replay Attack e Spoofing.

### 3.1.1 Descrizione

Il presente profilo assume l'esistenza di un trust tra fruitore ed erogatore, che permette il riconoscimento del certificato X.509, o la CA emittente dell'erogatore, così come previsto dal protocollo Transport Layer Security.

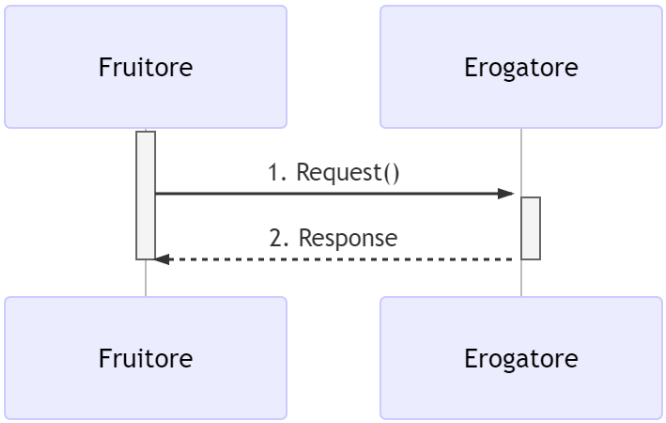


Figura 1 - Sicurezza di canale e/o Autenticazione dell'erogatore

La sequenza dei messaggi di richiesta/risposta avviene dopo aver instaurato il canale di trasmissione sicuro.

### 3.1.2 Regole di processamento

Il canale sicuro tra erogatore e fruitore viene instaurato utilizzando il protocollo TLS, secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

**A: Richiesta**

1. Il fruitore costruisce un messaggio di richiesta;
2. Il fruitore spedisce sul canale sicuro stabilito il messaggio di richiesta all'interfaccia di servizio dell'erogatore.

**B: Risposta**

3. L'erogatore elabora il messaggio e restituisce il risultato.

Come indicato in RFC 5246 l'impiego del protocollo TLS garantisce a livello di canale:

- l'autenticazione dell'erogatore identificato mediante il certificato X.509;
- la confidenzialità dei dati scambiati;
- l'integrità dei dati scambiati.

L'impiego del protocollo TLS mitiga il rischio di:

- Replay Attack;
- Spoofing.

## 3.2 [ID_AUTH_CHANNEL_02] Direct Trust mutual Transport-Level Security

Comunicazione tra fruitore ed erogatore che assicuri a livello di canale:

- confidenzialità;
- integrità;
- identificazione dell'erogatore e del fruitore, quali organizzazioni;
- difesa dalle minacce derivanti dagli attacchi: Replay Attack e Spoofing.

Il presente pattern è da applicare in maniera alternativa ai pattern di sicurezza indicati al capitolo "Accesso del soggetto fruitore" e solo nell'ipotesi in cui il fruitore non possa accreditarsi alla Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD.

Nel ricordare che l'applicazione del presente pattern PUÒ avvenire solo nel caso in cui il fruitore non possa accreditarsi alla Piattaforma Digitale Nazionale Dati per l'interoperabilità, si evidenza che entro 12 mesi dal superamento di tale impedimento l'erogatore e fruitore DEVONO aggiornare le modalità di costituzione del trust assicurando lo stesso per il tramite della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

### 3.2.1 Descrizione

Il presente profilo assume l'esistenza di un trust tra fruitore (client) ed erogatore (server), che permette il riconoscimento da entrambe le parti dei certificati X.509, o le CA emittenti, così come previsto dal protocollo Transport Layer Security.

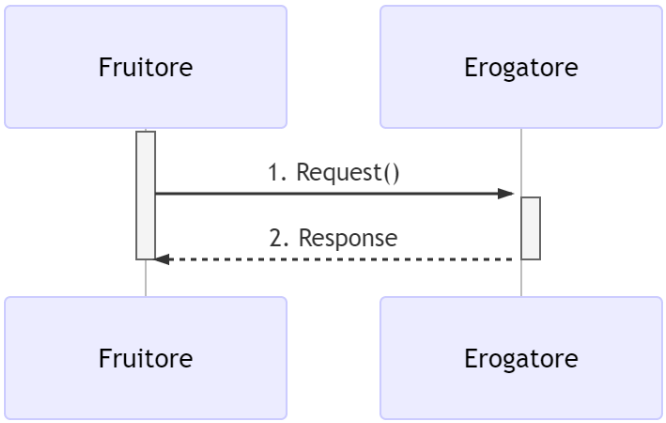


Figura 2 - Sicurezza di canale e/o Autenticazione delle organizzazioni

La sequenza dei messaggi di richiesta/risposta avviene dopo aver instaurato il canale di trasmissione sicuro.

### 3.2.2 Regole di processamento

Il canale sicuro tra erogatore e fruitore viene instaurato in mutua autenticazione utilizzando il protocollo TLS, secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

**A: Richiesta**

4. Il fruitore costruisce un messaggio di richiesta;
5. Il fruitore spedisce sul canale sicuro stabilito il messaggio di richiesta all'interfaccia di servizio dell'erogatore.

**B: Risposta**

6. L'erogatore elabora il messaggio e restituisce il risultato.

Come indicato in RFC 5246 l'impiego del protocollo TLS garantisce a livello di canale:

- l'autenticazione dell'erogatore e fruitore identificato mediante il certificato X.509;
- la confidenzialità dei dati scambiati;
- l'integrità dei dati scambiati.

L'impiego del protocollo TLS mitiga il rischio di:

- Replay Attack;
- Spoofing.

Capitolo 4

# Accesso del soggetto fruitore

Di seguito le indicazioni per le tecnologie accolte dal ModI.

L'AgID assicura l'aggiornamento degli stessi per soddisfare le esigenze espresse dalle PA.

## 4.1 [ID_AUTH_SOAP_01] Direct Trust con certificato X.509 su SOAP

Comunicazione tra fruitore ed erogatore che assicuri a livello di messaggio:

- accesso del soggetto fruitore, quale organizzazione o unità organizzativa fruitrice, o entrambe le parti.

### 4.1.1 Descrizione

Il presente profilo specializza lo standard OASIS Web Services Security X.509 Certificate Token Profile Versione 1.1.1.

Si assume l'esistenza di un trust tra fruitore ed erogatore, che permette il riconoscimento da parte dell'erogatore del certificato X.509, o la CA emittente.

Il meccanismo con cui è stabilito il trust, inclusa la modalità di scambio dei certificati X.509, non condiziona il presente profilo.

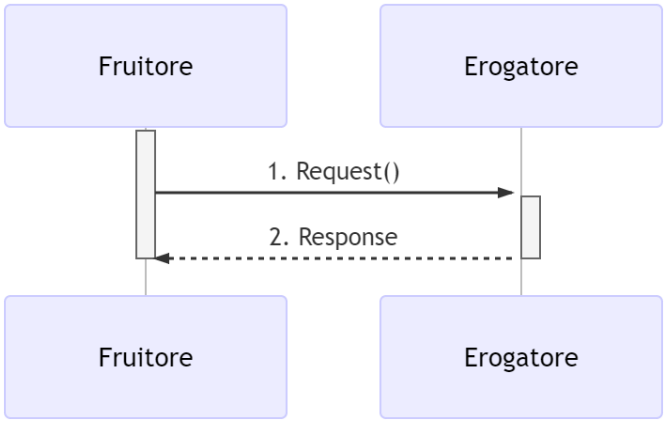


Figura 3 - Accesso del Fruitore

Il fruitore inoltra un messaggio all'interfaccia di servizio dell'erogatore includendo o referenziando il certificato X.509 e una porzione significativa del messaggio firmata.

L'erogatore, ricevuto il messaggio, verifica il certificato X.509 e valida la porzione firmata del messaggio. Se la verifica e la validazione sono superate, l'erogatore elabora la richiesta e produce la relativa risposta.

### 4.1.2 Regole di processamento

**A: Richiesta**

1. Il fruitore costruisce un messaggio SOAP per il servizio.
2. Il fruitore aggiunge al messaggio l'header WS-Addressing e l'elemento <wsu:Timestamp> composto dagli elementi <wsu:Created> e <wsu:Expires>
3. Il fruitore calcola la firma per gli elementi significativi del messaggio, in particolare <wsu:Timestamp> e <wsa:To> del blocco WS-Addressing. Il digest è firmato usando la chiave privata associata al certificato X.509 del fruitore. L'elemento <Signature> è posizionato nell'header <Security> del messaggio.
4. Il fruitore referenzia il certificato X.509 usando in maniera alternativa, nell'header <Security>, i seguenti elementi previsti nella specifica ws-security:
    a. <wsse:BinarySecurityToken>
    b. <wsse:KeyIdentifier>
    c. <wsse:SecurityTokenReference>
5. Il fruitore spedisce il messaggio all'interfaccia di servizio dell'erogatore.

**B: Risposta**

6. L'erogatore verifica il contenuto dell'elemento <wsu:Timestamp> nell'header del messaggio al fine di verificare la validità temporale del messaggio.
7. L'erogatore verifica la corrispondenza tra se stesso e quanto definito nell'elemento <wsa:To> del blocco WS-Addressing.
8. L'erogatore recupera il certificato X.509 referenziato nell'header <Security>.
9. L'erogatore verifica il certificato secondo i criteri del trust.
10. L'erogatore valida l'elemento <Signature> nell'header <Security>.
11. L'erogatore garantisce l'accesso al fruitore.
12. Se le azioni da 6 a 11 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato.

Note:

- In merito agli algoritmi da utilizzare nell'elemento <Signature> rispettivamente <DigestMethod>, <SignatureMethod> e <CanonicalizationMethod> si fa riferimento agli algoritmi indicati nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).
- Un meccanismo simile può essere utilizzato specularmente per l'erogatore.

### 4.1.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore relativo ad un servizio di echo.

I namespace utilizzati nel tracciato sono riportati di seguito:

```
soap="http://www.w3.org/2003/05/soap-envelope"
wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
ds="http://www.w3.org/2000/09/xmldsig#"
ec="http://www.w3.org/2001/10/xml-exc-c14n#"
```

```
        <?xml version="1.0"?>
        <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
          <soap:Header>
            <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd"
              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" soap:mustUnderstand="1">
              <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-soap-message-security-1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-x509-token-profile-1.0#X509v3" wsu:Id="X509-39011475-65d5-446e-ba38-
be84220fd720">MIICqDCCAZCgAwIBAgIEXLSSUTANBgkqhkiG9w0BAQsFADAW...</wsse:BinarySecurityToken>
              <wsu:Timestamp wsu:Id="TS-819df7b7-379d-48f7-8d9c-28c5b5d252f0">
                <wsu:Created>2019-04-15T14:53:34.649Z</wsu:Created>
                <wsu:Expires>2019-04-15T14:58:34.649Z</wsu:Expires>
              </wsu:Timestamp>
              <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-6e09e972-cbe6-43fc-a10c-
38e6dce56dbe">
                <ds:SignedInfo>
                  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                    <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap"/>
                  </ds:CanonicalizationMethod>
                  <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
                  <ds:Reference URI="#TS-819df7b7-379d-48f7-8d9c-28c5b5d252f0">
                    <ds:Transforms>
                      <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                        <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap    wsse"/>
                      </ds:Transform>
                    </ds:Transforms>
                    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                    <ds:DigestValue>K/3Fq1fYjG5PXv8UlKBuT4XBCWudGR5w2M10wPcZ/Yo=**</ds:DigestValue>
                  </ds:Reference>
                  <ds:Reference URI="#id-96f9b013-17e5-489d-8068-52c3f1345c75">
                    <ds:Transforms>
                      <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                        <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap"/>
                      </ds:Transform>
                    </ds:Transforms>
                    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                    <ds:DigestValue>eH3Vlc3l19NbBawDOuFDN11BfmbgGAnl6Z4LpJVM3UM=**</ds:DigestValue>
                  </ds:Reference>
                </ds:SignedInfo>
                <ds:SignatureValue>jAtZqkfRcFJW+jx9YDv+r2Q8V4IWEWLAZckZlWsmo...</ds:SignatureValue>
                <ds:KeyInfo Id="KI-32484d1e-867e-4465-a96f-52a8668d5a0c">
                  <wsse:SecurityTokenReference xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd"
                    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" wsu:Id="STR-3cf69cce-c56f-461a-905d-dfc20ab0742c">
                    <wsse:Reference URI="#X509-39011475-65d5-446e-ba38-be84220fd720"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
                  </wsse:SecurityTokenReference>
```

```
                </ds:KeyInfo>
              </ds:Signature>
            </wsse:Security>
            <Action
xmlns="http://www.w3.org/2005/08/addressing">http://profile.security.modi.agid.org/HelloWorld/sayHi</Action
>
            <MessageID xmlns="http://www.w3.org/2005/08/addressing">urn:uuid:55e6bc57-2286-4b7d-82a9-
fdbcf57721b1</MessageID>
            <To xmlns="http://www.w3.org/2005/08/addressing"
              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" wsu:Id="id-96f9b013-17e5-489d-8068-52c3f1345c75">https://api.ente.example/soap/echo/v1</To>
            <ReplyTo xmlns="http://www.w3.org/2005/08/addressing">
              <Address>http://www.w3.org/2005/08/addressing/anonymous</Address>
            </ReplyTo>
          </soap:Header>
          <soap:Body>
            <ns2:sayHi xmlns:ns2="http://profile.security.modi.agid.org/">
              <arg0>OK</arg0>
            </ns2:sayHi>
          </soap:Body>
        </soap:Envelope>
```

Il tracciato rispecchia le seguenti scelte implementative esemplificative:

- riferimento al security token (BinarySecurityToken)
- algoritmi di canonizzazione (CanonicalizationMethod)
- algoritmi di firma (SignatureMethod)
- algoritmo per il digest (DigestMethod)

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

## 4.2 [ID_AUTH_SOAP_02] Direct Trust con certificato X.509 su SOAP con unicità del token/messaggio

Il seguente profilo estende il profilo ID_AUTH_SOAP_01. Comunicazione tra fruitore ed erogatore che assicuri a livello di messaggio:

- accesso del soggetto fruitore, quale organizzazione o unità organizzativa fruitore, o entrambe le parti;
- difesa dalle minacce derivanti dagli attacchi: Replay Attack.

### 4.2.1 Descrizione

Il presente profilo specializza lo standard OASIS Web Services Security X.509 Certificate Token Profile Versione 1.1.1.

Si assume l'esistenza di un trust tra fruitore ed erogatore, che permette il riconoscimento da parte dell'erogatore del certificato X.509, o la CA emittente.

Il meccanismo con cui è stabilito il trust, inclusa la modalità di scambio dei certificati X.509, non condiziona il presente profilo.

Il fruitore inoltra un messaggio all'interfaccia di servizio dell'erogatore includendo o referenziando il certificato X.509 e assicurando la firma dei claim del messaggio.

L'erogatore, ricevuto il messaggio, verifica il certificato X.509, valida la firma dei claim e garantisce l'accesso al fruitore. Se la verifica e la validazione sono superate, l'erogatore elabora la richiesta e produce la relativa risposta.

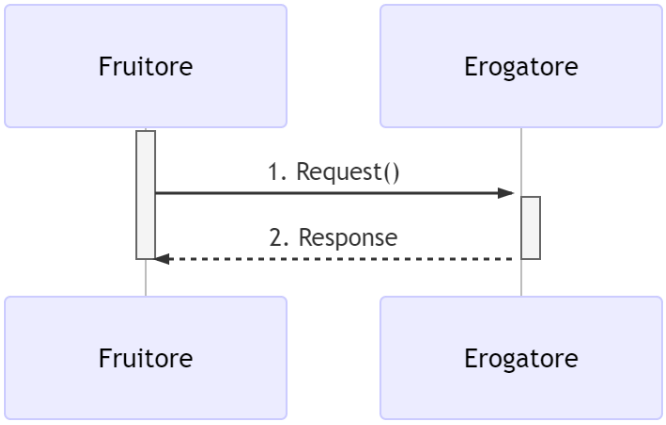


Figura 4 - Accesso del Fruitore

### 4.2.2 Regole di processamento

**A: Richiesta**

1. Il fruitore costruisce un messaggio SOAP per il servizio.
2. Il fruitore aggiunge al messaggio l'header WS-Addressing e l'elemento <wsu:Timestamp> composto dagli elementi <wsu:Created> e <wsu:Expires>
3. Il fruitore calcola la firma per gli elementi significativi del messaggio, in particolare <wsa:To> e <wsa:MessageID> del blocco WS-Addressing e <wsu:Timestamp>. Il digest è firmato usando la chiave privata associata al certificato X.509 del fruitore. L'elemento <Signature> è posizionato nell'header <Security> del messaggio.
4. Il fruitore referenzia il certificato X.509 usando in maniera alternativa, nell'header <Security>, i seguenti elementi previsti nella specifica ws-security:
    a. <wsse:BinarySecurityToken>
    b. <wsse:KeyIdentifier>
    c. <wsse:SecurityTokenReference>
5. Il fruitore spedisce il messaggio all'interfaccia di servizio dell'erogatore.

B: Risposta

6. L'erogatore verifica il contenuto dell'elemento <wsu:Timestamp> nell'header del messaggio al fine di verificare la validità temporale del messaggio.
7. L'erogatore verifica la corrispondenza tra se stesso e quanto definito nell'elemento <wsa:To> del blocco WS-Addressing.
8. L'erogatore verifica l'univocità del <wsa:MessageID> del blocco WS-Addressing
9. L'erogatore recupera il certificato X.509 referenziato nell'header <Security>.
10. L'erogatore verifica il certificato secondo i criteri del trust.
11. L'erogatore valida l'elemento <Signature> nell'header <Security>.
12. L'erogatore garantisce l'accesso al fruitore.
13. Se le azioni da 6 a 12 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato.

Note:

- In merito agli algoritmi da utilizzare nell'elemento <Signature> rispettivamente <DigestMethod>, <SignatureMethod> e <CanonicalizationMethod> si fa riferimento agli algoritmi indicati nelle Linee Guida sulla sicurezza, emanate

dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

- Un meccanismo simile può essere utilizzato specularmente per l'erogatore.

### 4.2.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore relativo ad un servizio di echo.

I namespace utilizzati nel tracciato sono riportati di seguito:

```
soap="http://schemas.xmlsoap.org/soap/envelope/"
wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
ds="http://www.w3.org/2000/09/xmldsig#"
ec="http://www.w3.org/2001/10/xml-exc-c14n#"
```

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" soap:mustUnderstand="1">
      <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" wsu:Id="X509-bf881daf-371a-4d18-9502-d9f92af9a949">MIICqDCCAZ
                CgAwIBAgIEXLSSUTANBgkqhkiG9w0BAQsFADAWMRQwEgYDVQQDDAttb2RpU2VjUHJvZjA
                eFw0xOTA0MTUxNDE2NDlaFw0yNDA0MTUxNDE2NDlaMBYxFDASBgNVBAMMC21vZGlTZWNQ
                cm9mMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvBJNKBiLS+ZcmwGUku5l2
                FKeHogeSZejjOOrO2Ag6DGPXo1MtHt2XwgUXmgT+v0IjhZp5XH2XbwSWw2EMWSG3Zz0CJ
                ILqWGPg0M/LxaIZAxSdxJpVNWg/profO+xKz0B6QHK+I0yecHg7TtI4es9AuDyR4pKslp
                cXyMEqJQ7m5N8v2e4WldeHF2SRN/ereEOuewEi15c7akh4TdkGdiwOSif2AXIugHRgdpH
                jH86iJxFu24IJmBA7C5tytz7mfKollGhI9+2d0902ayVshCV4/pmnX0pDiGayV1C6SDPT
                bapXKXJrp1+fBHaUkDY+W/2Q9sC4o8pttmcpHeMRxFDkwIDAQABMA0GCSqGSIb3DQEBCw
                UAA4IBAQALwKbIm8S2BpYpHaqMwJLeWBPCaDeT7J+KDj39Ac3YxDb8E/hGM+Hn1mq2ssY
                qu5JTvuAQ9o8v3UpcIct15RPgOKYfBzxnH1h2vCavpiFCFTc6UoQgPBZGyyNOOKNOxEnX
                tW7ff1gl2GRLIWXlXDf1fdX7VJVBqWfBvivhIbsDa5LRBCrNsXORx2azUb5QBgMm2UZJx
                YA3+dFRgYmLSY/RyRLf0o03lwCRhAyrU7ya9IMYgrxgjEos2fHB2IGJJ1Wh+gTQWMP+wJ
                ymlC0qyjTHx5pyZOzJGtH5HnaVU7EgtxdBRC9dTlWVpNgmD8nS6Yr/am5cZJZrkIHRyfx
                qkA2W
      </wsse:BinarySecurityToken>
      <wsu:Timestamp wsu:Id="TS-09f1357c-beb4-4804-9410-76c5a06e2e48">
        <wsu:Created>2019-04-15T15:02:15.515Z</wsu:Created>
        <wsu:Expires>2019-04-15T15:07:15.515Z</wsu:Expires>
```

```
            </wsu:Timestamp>
            <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-4d949c5b-968b-4fd5-be67-
4cd1d1a41ce3">
              <ds:SignedInfo>
                <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                  <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap"/>
                </ds:CanonicalizationMethod>
                <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
                <ds:Reference URI="#TS-09f1357c-beb4-4804-9410-76c5a06e2e48">
                  <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                      <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap wsse"/>
                    </ds:Transform>
                  </ds:Transforms>
                  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                  <ds:DigestValue>HPYjNXdxIuJIWk1EArE+8PIgyWt5nAD+upwcjOSDB20=</ds:DigestValue>
                </ds:Reference>
                <ds:Reference URI="#id-27c23bc8-0c4f-4d98-b046-6e590ea9661b">
                  <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                      <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap"/>
                    </ds:Transform>
                  </ds:Transforms>
                  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                  <ds:DigestValue>MJzRD4ZRMsFOxskbnfNV9BnDTCLxuLSnmZ8I4IjaxHw=</ds:DigestValue>
                </ds:Reference>
                <ds:Reference URI="#id-fb4c1fa0-e804-4169-b70e-5b55c5f9d912">
                  <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                      <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"
PrefixList="soap"/>
                    </ds:Transform>
                  </ds:Transforms>
                  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                  <ds:DigestValue>MIi+ovLTqYu1HqxUtmUnuhVdMmNKOpOX8vn/fKjvQFU=</ds:DigestValue>
                </ds:Reference>
              </ds:SignedInfo>
              <ds:SignatureValue>SBYs6aikHbfsHHV04ifV/ljVTysxNLRTPU6gsOGJamWG
                          YLMPqOETjBf+NFJhPDVdolQSSHw0SD7uA/RlYkE9amRH1K+hoaUIa/PEhPgC1io/LqZdi
                          3rt+b8uRlk+CXcUKOObgf/N960F/sM6s0ArKQxg/Yx6pqWamXBXo0PH/1FvHSgwdA62s0
                          +Sli96qY0EnJPoyKIrqzskiscLXI1jCe8sesyA+xtJ0qBdFKAn2af48sVStPFv4gizC8+
                          bsCRpQ36ihUIlI8DInJ13EgoKV9/rC4PheExO7HvSNTpBFdQt+Wr9wAb3oHq4urRBdugA
6mX2xaJ8/XyZVajivvuVTw==
              </ds:SignatureValue>
              <ds:KeyInfo Id="KI-dab2ce54-b000-439a-bcc2-9b8249626a1c">
                <wsse:SecurityTokenReference xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd"
                  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" wsu:Id="STR-068909fe-1a64-4cf1-bd5a-355a20b0495f">
```

```
                    <wsse:Reference URI="#X509-bf881daf-371a-4d18-9502-d9f92af9a949"
Valuewss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
                  </wsse:SecurityTokenReference>
                </ds:KeyInfo>
              </ds:Signature>
            </wsse:Security>
            <Action
xmlns="http://www.w3.org/2005/08/addressing">http://profile.security.modi.agid.org/HelloWorld/sayHi
            </Action>
            <MessageID xmlns="http://www.w3.org/2005/08/addressing"
              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" wsu:Id="id-fb4c1fa0-e804-4169-b70e-5b55c5f9d912">urn:uuid:46d
                    a4ec1-f962-4f24-8524-48bb74b505d7
            </MessageID>
            <To xmlns="http://www.w3.org/2005/08/addressing"
              xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" wsu:Id="id-27c23bc8-0c4f-4d98-b046-6e590ea9661b">http://local
                    host:8080/security-profile/echo
            </To>
            <ReplyTo xmlns="http://www.w3.org/2005/08/addressing">
              <Address>http://www.w3.org/2005/08/addressing/anonymous
              </Address>
            </ReplyTo>
          </soap:Header>
          <soap:Body>
            <ns2:sayHi xmlns:ns2="http://profile.security.modi.agid.org/">
              <arg0>OK</arg0>
            </ns2:sayHi>
          </soap:Body>
        </soap:Envelope>
```

Il tracciato rispecchia le seguenti scelte implementative esemplificative:

- riferimento al security token (BinarySecurityToken)
- algoritmi di canonizzazione (CanonicalizationMethod)
- algoritmi di firma (SignatureMethod).
- algoritmo per il digest (DigestMethod)

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

## 4.3 [ID_AUTH_REST_01] Direct Trust con certificato X.509 su REST

Comunicazione tra fruitore ed erogatore che assicuri a livello di messaggio:

- accesso del soggetto fruitore, quale organizzazione o unità organizzativa fruitore, o entrambe le parti.

### 4.3.1 Descrizione

Il presente profilo declina l'utilizzo di:

- JSON Web Token (JWT) definita dall'RFC 7519
- JSON Web Signature (JWS) definita dall'RFC 7515

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, tramite il materiale crittografico depositato applicando i profili di emissione dei voucher previsti dalla stessa.

La costituzione del trust tra fruitore ed erogatore PUÒ essere realizzata al di fuori della Piattaforma Digitale Nazionale Dati per l'interoperabilità, attraverso l'utilizzo di materiale crittografico basato su certificati X.509, solo nel caso in cui il fruitore non possa accreditarsi alla stessa e comunque entro 12 mesi dal superamento di tale impedimento l'erogatore e fruitore DEVONO aggiornare le modalità di costituzione del trust assicurando lo stesso per il tramite della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

In quanto segue si declina il presente pattern in assenza della Piattaforma Digitale Nazionale Dati per l'interoperabilità, si rimanda alle "Linee Guida sull'infrastruttura tecnologica della Piattaforma Digitale Nazionale Dati per l'interoperabilità dei sistemi informativi e delle basi di dati" adottate ai sensi dell'articolo 50-ter, comma 2 del CAD l'esplicitazione delle modalità da applicare in presenza della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

Il fruitore inoltra un messaggio all'erogatore includendo o referenziando il certificato X.509 e una porzione significativa del messaggio firmata.

L'erogatore, ricevuto il messaggio, verifica il certificato X.509 e valida la porzione firmata del messaggio, inclusa la corrispondenza del destinatario e l'intervallo di validità della firma. Se la verifica e la validazione sono superate, l'erogatore elabora la richiesta e produce la relativa risposta.

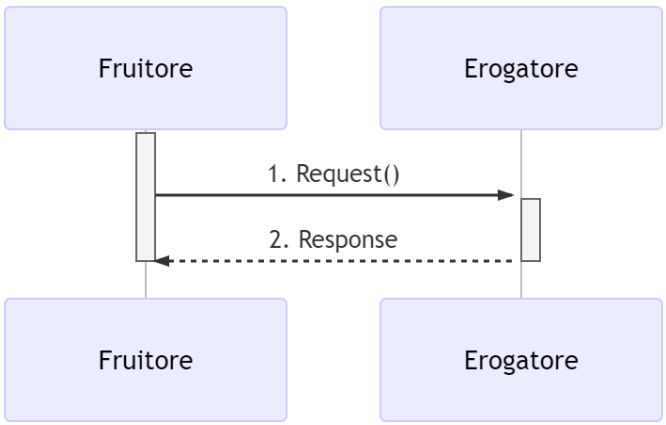


Figura 5 - Accesso del Fruitore

### 4.3.2 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il payload del messaggio (ad esempio un oggetto JSON)
2. Il fruitore, o il soggetto individuato dal trust definito tra erogatore e fruitore, costruisce il JWT popolando:
    a. il JOSE Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725
        ii. typ uguale a JWT
        iii. una o più delle seguenti opzioni per referenziare il certificato X.509:
            1. x5u (X.509 URL)
            2. x5c (X.509 Certificate Chain)
            3. x5t#S256 (X.509 Certificate SHA-256 Thumbprint)

b. il payload del JWT coi claim rappresentativi degli elementi chiave del messaggio, contenente almeno:
    i. i riferimenti temporali di emissione e scadenza: iat, exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf.
    ii. il riferimento dell'erogatore in aud

3. il fruitore, o il soggetto individuato dal trust definito tra erogatore e fruitore, firma il token adottando la JWS Compact Serialization
4. il fruitore posiziona il JWT nell' HTTP header Authorization
5. Il fruitore spedisce il messaggio all'erogatore

**B: Risposta**

6. L'erogatore decodifica il JWT presente in HTTP header Authorization secondo le indicazioni contenute in RFC 7515#section-5.2, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel JOSE Header, in particolare verifica:
7. il contenuto dei claim iat ed exp;
8. la corrispondenza tra se stesso e il claim aud;
9. L'erogatore recupera il certificato X.509 referenziato nel JOSE Header facendo attenzione alle indicazioni contenute in RFC 8725#section-3.10
10. L'erogatore verifica il certificato secondo i criteri del trust
11. L'erogatore valida la firma verificando l'elemento Signature del JWT
12. L'erogatore garantisce l'accesso al fruitore
13. Se le azioni da 6 a 10 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato

Note:

- Gli algoritmi da utilizzare in alg sono indicati nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).
- Un meccanismo simile può essere utilizzato specularmente per l'erogatore.

- Per prevenire il rischio di user-enumeration, i messaggi di errore di autenticazione non DEVONO fornire informazioni sull'esistenza o meno dell'utenza.

### 4.3.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'erogatore.

Esempio porzione messaggio HTTP.

```
GET https://api.erogatore.example/rest/service/v1/hello/echo/Ciao HTTP/1.1
Accept: application/json
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5c.vz8...
```

Esempio porzione JWT

```
# header

{
"alg": "ES256",
"typ": "JWT",
"x5c": [
"MIICyzCCAbOgAwIBAgIEC..."
]
}

# *payload*
{
"iat": 1554382877,
"nbf": 1554382877,
"exp": 1554382879,
"iss": "https://api.fruitore.example",
"sub": "https://api.fruitore.example",
"aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
}
```

Gli elementi presenti nel tracciato rispettano le seguenti scelte implementative e includono:

- l'intervallo temporale di validità, in modo che il JWT possa essere usato solo tra gli istanti nbf ed exp;

- indica l'istante iat di emissione del JWT. Se le parti possono accordarsi nel considerarlo come l'istante iniziale di validità del token, RFC 7519 non assegna a questo claim nessun ruolo specifico nella validazione, a differenza di nbf;
- il riferimento al firmatario del token nel claim aggiuntivo iss, che deve essere raccordato con il contenuto del certificato;
- il riferimento al fruitore nel claim aggiuntivo sub;
- il destinatario del JWT, che DEVE sempre essere validato;
- contenuto della certificate chain X.509 (x5c)
- algoritmi di firma e digest (alg).

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

## 4.4 [ID_AUTH_REST_02] Direct Trust con certificato X.509 su REST con unicità del token/messaggio

Il seguente profilo estende il profilo ID_AUTH_REST_01. Comunicazione tra fruitore ed erogatore che assicuri a livello di messaggio:

- accesso del soggetto fruitore, quale organizzazione o unità organizzativa fruitore, o entrambe le parti
- la difesa dalle minacce derivanti dagli attacchi: Replay Attack quando il JWT o il messaggio NON DEVONO essere riprocessati.

### 4.4.1 Descrizione

Il presente profilo declina l'utilizzo di:

- JSON Web Token (JWT) definita dall'RFC 7519
- JSON Web Signature (JWS) definita dall'RFC 7515

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, tramite il materiale crittografico depositato applicando i profili di emissione dei voucher previsti dalla stessa.

La costituzione del trust tra fruitore ed erogatore PUÒ essere realizzata al di fuori della Piattaforma Digitale Nazionale Dati per l'interoperabilità, attraverso l'utilizzo di materiale crittografico basato su certificati X.509, solo nel caso in cui il fruitore non possa accreditarsi alla stessa e comunque entro 12 mesi dal superamento di tale impedimento l'erogatore e fruitore DEVONO aggiornare le modalità di costituzione del trust assicurando lo stesso per il tramite della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

In quanto segue si declina il presente pattern in assenza della Piattaforma Digitale Nazionale Dati per l'interoperabilità, si rimanda alle "Linee Guida sull'infrastruttura tecnologica della Piattaforma Digitale Nazionale Dati per l'interoperabilità dei sistemi informativi e delle basi di dati" adottate ai sensi dell'articolo 50-ter, comma 2 del CAD l'esplicitazione delle modalità da applicare in presenza della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

Il fruitore inoltra un messaggio all'erogatore includendo o referenziando il certificato X.509 e una porzione significativa del messaggio firmata.

L'erogatore, ricevuto il messaggio, verifica il certificato X.509 e valida la porzione firmata del messaggio, inclusa la corrispondenza del destinatario e l'intervallo di validità della firma. Se la verifica e la validazione sono superate, l'erogatore elabora la richiesta e produce la relativa risposta.

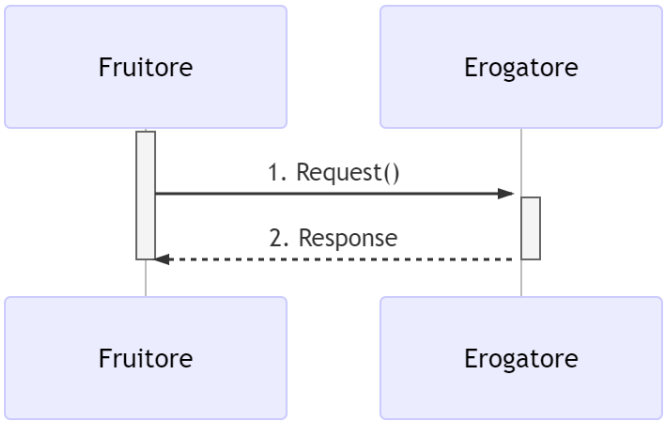


Figura 6 - Accesso del Fruitore

### 4.4.2 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il payload del messaggio (ad esempio un oggetto JSON)
2. Il fruitore, o il soggetto individuato dal trust definito tra erogatore e fruitore, costruisce il JWT popolando:
    a. il Jose Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725
        ii. typ uguale a JWT
        iii. una o più delle seguenti opzioni per referenziare il certificato X.509:
            1. x5u (X.509 URL)
        iv. x5c (X.509 Certificate Chain)
        v. x5t#S256 (X.509 Certificate SHA-256 Thumbprint)
    b. il payload del JWT coi claim rappresentativi degli elementi chiave del messaggio, contenente almeno:
        i. i riferimenti temporali di emissione e scadenza: iat, exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf.
        ii. il riferimento dell'erogatore in aud;

iii. un identificativo univoco del token jti. Se utile alla logica applicativa l'identificativo può essere anche collegato al messaggio.

3. il fruitore, o il soggetto individuato dal trust definito tra erogatore e fruitore firma il token adottando la JWS Compact Serialization
4. il fruitore posiziona il JWT nell' HTTP header Authorization
5. Il fruitore spedisce il messaggio all'erogatore.

**B: Risposta**

6. L'erogatore decodifica il JWT presente in HTTP header Authorization secondo le indicazioni contenute in RFC 7515#section-5.2, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel JOSE Header, in particolare verifica:
   a. il contenuto dei claim iat ed exp;
   b. la corrispondenza tra se stesso e il claim aud;
   c. l'univocità del claim jti
7. L'erogatore recupera il certificato X.509 referenziato nel JOSE Header facendo attenzione alle indicazioni contenute in RFC 8725#section-3.10
8. L'erogatore verifica il certificato secondo i criteri del trust
9. L'erogatore valida la firma verificando l'elemento Signature del JWT
10. L'erogatore garantisce l'accesso al fruitore
11. Se le azioni da 6 a 10 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato.

Note:

- In merito agli algoritmi da utilizzare si fa riferimento alle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).
- Un meccanismo simile può essere utilizzato specularmente per l'erogatore.

### 4.4.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

Esempio porzione pacchetto HTTP.

```
GET https://api.erogatore.example/rest/service/v1/hello/echo/Ciao HTTP/1.1
Accept: application/json
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5c.vz8...
```

Esempio porzione JWT

```
# *header*
{
        "alg": "ES256",
        "typ": "JWT",
        "x5c": [
                "MIICyzCCAbOgAwIBAgIEC..."
                ]
}

# *payload*
{
        "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
        "iat": 1516239022,
        "nbf": 1516239022,
        "exp": 1516239024,
        "iss": "https://api.fruitore.example",
        "sub": "https://api.fruitore.example",
        "jti": "065259e8-8696-44d1-84c5-d3ce04c2f40d"
}
```

Gli elementi presenti nel tracciato rispettano le seguenti scelte implementative e includono:

- l'intervallo temporale di validità, in modo che il JWT possa essere usato solo tra gli istanti nbf ed exp;
- indica l'istante iat di emissione del JWT. Se le parti possono accordarsi nel considerarlo come l'istante iniziale di validità del token, RFC 7519 non assegna a questo claim nessun ruolo specifico nella validazione, a differenza di nbf;
- il riferimento al firmatario del token nel claim aggiuntivo iss, che deve essere raccordato con il contenuto del certificato;
- il riferimento al fruitore nel claim aggiuntivo sub;

- il destinatario del JWT, che DEVE sempre essere validato;
- contenuto della certificate chain X.509 (x5c)
- algoritmi di firma e digest (alg).

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

# Capitolo 5

# Integrità

Di seguito le indicazioni per le tecnologie accolte dal ModI.

L'AgID assicura l'aggiornamento degli stessi per soddisfare le esigenze espresse dalle PA.

## 5.1 [INTEGRITY_SOAP_01] Integrità del payload del messaggio SOAP

Il presente profilo estende ID_AUTH_SOAP_01 o ID_AUTH_SOAP_02, aggiungendo alla comunicazione tra fruitore ed erogatore a livello di messaggio:

- integrità del payload del messaggio.

### 5.1.1 Descrizione

Il presente profilo specializza lo standard OASIS Web Services Security X.509 Certificate Token Profile Versione 1.1.1.

Si assume l'esistenza di un trust tra fruitore ed erogatore, che permette il riconoscimento da parte dell'erogatore del certificato X.509, o la CA emittente.

Il meccanismo con cui è stabilito il trust non condiziona il presente profilo.

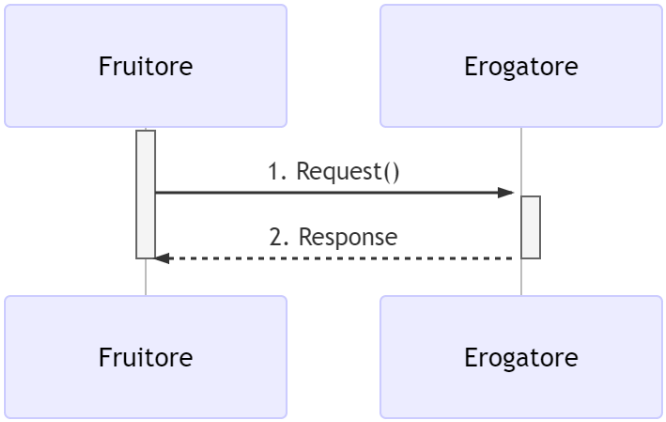


Figura 7 - Integrità del payload del messaggio

Il fruitore inoltra un messaggio all'interfaccia di servizio dell'erogatore includendo o referenziando il certificato X.509 e la firma del payload del messaggio.

L'erogatore, ricevuto il messaggio, verifica il certificato X.509 e valida l'integrità del payload del messaggio firmato. Se la verifica e la validazione sono superate, l'erogatore elabora la richiesta e produce la relativa risposta.

### 5.1.2 Regole di processamento

**A: Richiesta**

1. Il fruitore costruisce un messaggio SOAP per il servizio.
2. Il fruitore calcola la firma del payload del messaggio usando l'XML Signature. Il digest è firmato usando la chiave privata associata al certificato X.509 del fruitore. L'elemento <Signature> è posizionato nell'header <Security> del messaggio.
3. Il fruitore referenzia il certificato X.509 usando in maniera alternativa, nell'header <Security>, i seguenti elementi previsti nella specifica ws-security:
    a. <wsse:BinarySecurityToken>
    b. <wsse:KeyIdentifier>
    c. <wsse:SecurityTokenReference>
4. Il fruitore spedisce il messaggio all'interfaccia di servizio dell'erogatore.

**B: Risultato**

5. L'erogatore recupera il certificato X.509 referenziato nell'header <Security>.
6. L'erogatore verifica il certificato secondo i criteri del trust.

7. L'erogatore valida la firma verificando l'elemento <Signature> nell'header <Security>.
8. Se il certificato è valido anche per identificare il soggetto fruitore, l'erogatore autentica lo stesso
9. Se le azioni da 5 a 8 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato

Note:

- Per quanto riguarda gli algoritmi da utilizzare nell'elemento <Signature> rispettivamente <DigestMethod> , <SignatureMethod> e <CanonicalizationMethod> si fa riferimento agli algoritmi indicati nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).
- Un meccanismo simile può essere utilizzato per garantire l'integrità del payload del messaggio risposta dell'erogatore al fruitore.

### 5.1.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

I namespace utilizzati nel tracciato sono riportati di seguito:

```
soap="http://schemas.xmlsoap.org/soap/envelope/"
wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
ds="http://www.w3.org/2000/09/xmldsig#"
ec="http://www.w3.org/2001/10/xml-exc-c14n#"
```

```
<?xml version="1.0"?>
<soap:Envelope>
  <soap:Header>
    <wsse:Security soap:mustUnderstand="1">
      <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-soap-message-security-1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-x5 09-token-profile-1.0#X509v3" wsu:Id="X509-44680ddc-e35a-4374-bcbf-2b6dcba722d7">MIICyzCCAb
OgAwIBAgIECxY+9TAhkiG9w...</wsse:BinarySecurityToken>
```

```
            <ds:Signature Id="SIG-f58c789e-e3d3-4ec3-9ca7-d1e9a4a90f90">
              <ds:SignedInfo>
                <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                  <ec:InclusiveNamespaces PrefixList="soap"/>
                </ds:CanonicalizationMethod>
                <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
                <ds:Reference URI="#bd-567d101-aed1-789e-81cb-5ae1c5dbef1a">
                  <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                      <ec:InclusiveNamespaces PrefixList="soap"/>
                    </ds:Transform>
                  </ds:Transforms>
                  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                  <ds:DigestValue>0cJNCJ1W8Agu66fGTX1PRyy0EUNUQ9OViFlm8qf8Ysw=* *</ds:DigestValue>
                </ds:Reference>
              </ds:SignedInfo>
              <ds:SignatureValue>AIrDa7ukDfFJD867goC+c7K3UampxpX/Nj/...</ds:SignatureValue>
              <ds:KeyInfo Id="KI-cad9ee47-dec8-4340-8fa1-74805f7e26f8">
                <wsse:SecurityTokenReference wsu:Id="STR-e193f25f-9727-4197-b7aa-25b01c9f2ba3">
                  <wsse:Reference URI="#X509-44680ddc-e35a-4374-bcbf-2b6dcba722d7"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
                </wsse:SecurityTokenReference>
              </ds:KeyInfo>
            </ds:Signature>
          </wsse:Security>
        </soap:Header>
        <soap:Body xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd" wsu:id="bd-567d101-aed1-789e-81cb-5ae1c5dbef1a">
          <ns2:sayHi xmlns:ns2="http://example.profile.security.modi.agid.gov.it/">
            <arg0>Hello World!</arg0>
          </ns2:sayHi>
        </soap:Body>
      </soap:Envelope>
```

Il codice rispecchia alcune scelte implementative esemplificative in merito:

- riferimento al security token (BinarySecurityToken)
- algoritmi di canonizzazione (CanonicalizationMethod)
- algoritmi di firma (SignatureMethod)
- algoritmo per il digest (DigestMethod)

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

## 5.2 [INTEGRITY_REST_01] Integrità del payload messaggio REST

Il presente profilo estende ID_AUTH_REST_01 o ID_AUTH_REST_02, aggiungendo alla comunicazione tra fruitore ed erogatore a livello di messaggio:

- integrità del payload del messaggio

Si adottano le indicazioni riportate in RFC 7231. Considereremo sempre richieste e risposte complete, con i metodi standard definiti in RFC 7231#section-4.

Questo scenario non copre quindi Range Requests RFC 7233 o HTTP method PATCH che trasmette una rappresentazione parziale.

### 5.2.1 Descrizione

Il presente profilo propone l'utilizzo di:

- semantica HTTP RFC 7231;
- Digest HTTP header RFC 3230 per l'integrità della rappresentazione della risorsa;
- JSON Web Token (JWT) definita dall' RFC 7519;
- JSON Web Signature (JWS) definita dall' RFC 7515.

Si assume l'esistenza di un trust tra fruitore ed erogatore, che permette il riconoscimento da parte dell'erogatore del certificato X.509, o la CA emittente.

Il meccanismo con cui è stabilito il trust non condiziona il presente profilo.

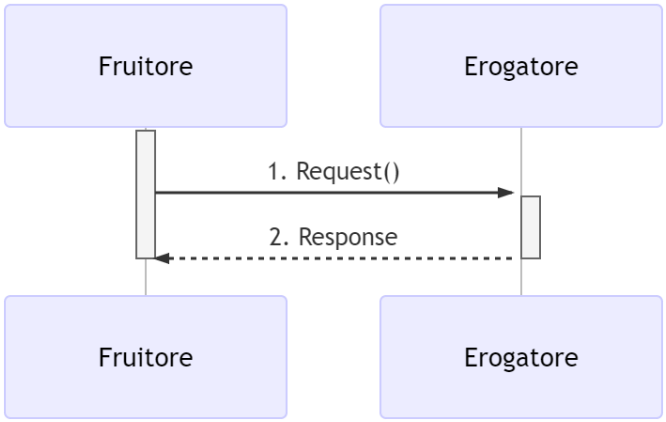


Figura 8 - Integrità del payload del messaggio

### 5.2.2 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il body del messaggio (ad esempio un oggetto JSON)
2. Il fruitore calcola il valore del Digest header dei representation data secondo le indicazioni in RFC 3230
3. Il fruitore individua l'elenco degli HTTP Header da firmare, incluso Digest e se presenti HTTP header Content-Type e HTTP header Content-Encoding
4. Il fruitore crea la struttura o la stringa da firmare in modo che includa gli http header da proteggere, i riferimenti temporali di validità della firma e degli estremi della comunicazione, ovvero:
    a. il JOSE Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725
        ii. typ uguale a JWT
        iii. una o più delle seguenti opzioni per referenziare il certificato X.509:
            1. x5u (X.509 URL)
            2. x5c (X.509 Certificate Chain)
            3. x5t#S256 (X.509 Certificate SHA-256 Thumbprint)
    b. i seguenti claim obbligatori:

i. i riferimenti temporali di emissione e scadenza: iat , exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf.
ii. il riferimento dell'erogatore in aud;

c. i seguenti claim, secondo la logica del servizio:
i. sub: oggetto (principal see RFC 3744#section-2) dei claim contenuti nel jwt
ii. iss: identificativo del mittente
iii. jti: identificativo del JWT, per evitare replay attack

d. il claim signed_headers con gli header http da proteggere ed i rispettivi valori, ovvero:
i. HTTP header Digest
ii. HTTP header Content-Type
iii. HTTP header Content-Encoding

5. il fruitore firma il token adottando la JWS Compact Serialization
6. il fruitore posiziona il JWS nell'header Agid-JWT-Signature
7. Il fruitore spedisce il messaggio all'erogatore.

**B: Risultato**

8. L'erogatore decodifica il JWS presente in Agid-JWT-Signature header secondo le indicazioni contenute in RFC 7515#section-5.2, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel Jose Header, in particolare verifica:
a. il contenuto dei claim iat , exp;
b. la corrispondenza tra se stesso e il claim aud;
c. l'univocità del claim jti se presente.
9. L'erogatore recupera il certificato X.509 referenziato nel JOSE Header facendo attenzione alle indicazioni contenute in RFC 8725#section-3.10
10. L'erogatore verifica il certificato secondo i criteri del trust
11. L'erogatore valida la firma verificando l'elemento Signature del JWS

12. L'erogatore verifica la corrispondenza tra i valori degli header passati nel messaggio e quelli presenti nel claim signed-header, Content-Type e Content-Encoding se presenti devono essere sempre firmati, come indicato nel punto A3
13. L'erogatore quindi verifica la corrispondenza tra Digest ed il payload body ricevuto
14. Se le azioni da 8 a 13 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato.

Note:

- Per gli algoritmi da utilizzare in alg e Digest si vedano le Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).
- Un meccanismo simile può essere utilizzato per garantire l'integrità della risposta da parte dell'erogatore al fruitore. In questo caso si ricorda che Digest fa riferimento al checksum del payload body della selected representation. Per una richiesta con HTTP method HEAD il server DEVE ritornare il checksum dell'ipotetico payload body ritornato dalla corrispondente richiesta con HTTP method GET.

### 5.2.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

Richiesta HTTP con Digest e representation metadata

```
POST https://api.erogatore.example/rest/service/v1/hello/echo/ HTTP/1.1
Accept: application/json
Agid-JWT-Signature: eyJhbGciOiJSUzI1NiIsInR5c.vz8...
Digest: SHA-256=cFfTOCesrWTLVzxn8fmHl4AcrUs40Lv5D275FmAZ96E=
Content-Type: application/json

{"testo": "Ciao mondo"}
```

Porzione JWS con campi protetti dalla firma

```
# *header*
{
```

```
  "alg": "ES256",
  "typ": "JWT",
  "x5c": [
        "MIICyzCCAbOgAwIBAgIEC..."
  ]
}
# *payload*

{
  "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
  "iat": 1516239022,
  "nbf": 1516239022,
  "exp": 1516239024,
  "signed headers": [
    {"digest": "SHA-256=cFfTOCesrWTLVzxn8fmHl4AcrUs40Lv5D275FmAZ96E="},
    {"content-type": "application/json"}
  ],
}
```

Il tracciato rispecchia alcune scelte implementative esemplificative in merito:

- include tutti gli elementi del JWS utilizzati in ID_AUTH_REST_02
- mette in minuscolo i nomi degli header firmati
- utilizza il claim custom signed_headers contenente una lista di json objects per supportare la firma di più header ed eventualmente verificare il loro ordinamento

Le parti, in base alle proprie esigenze, individuano gli specifici algoritmi secondo quanto indicato nelle Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

## 5.3 [INTEGRITY_REST_02] Integrità del payload delle request REST in PDND

Il presente pattern, nel contesto di Piattaforma Digitale Nazionale Dati per l'interoperabilità (di cui al comma 2 dell'articolo 50-ter del CAD), aggiunge alla comunicazione tra fruitore ed erogatore a livello di messaggio:

- integrità del payload della request del fruitore.

Si adottano le indicazioni riportate in RFC 7231. Considereremo sempre richieste e risposte complete, con i metodi standard definiti in RFC 7231#section-4.

Questo pattern non copre quindi Range Requests RFC 7233 o HTTP method PATCH che trasmette una rappresentazione parziale.

Il presente pattern DEVE essere applicato nel caso in cui fruitore ed erogatore risultano aderenti della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

L'integrità del payload della request del Fruitore, garantita attraverso l'applicazione del presente pattern, è riconosciuta nel perimetro degli aderenti alla Piattaforma Digitale Nazionale Dati per l'interoperabilità.

### 5.3.1 Descrizione

Il presente profilo propone l'utilizzo di:

- semantica HTTP RFC 7231;
- Digest HTTP header RFC 3230 per l'integrità della rappresentazione della risorsa;
- JSON Web Token (JWT) definita dall' RFC 7519;
- JSON Web Signature (JWS) definita dall' RFC 7515.

Si assume che il trust tra fruitore ed erogatore è costruito per il tramite di Piattaforma Digitale Nazionale Dati per l'interoperabilità che rende disponibile il materiale crittografico, nello specifico la chiave pubblica dichiarata dal fruitore relativamente al client da esso utilizzato per invocare l'e-service dell'erogatore.

In merito all'identificativo della chiave pubblica (kid) associata alla chiave privata utilizzata dal client, lo stesso è generato dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità.

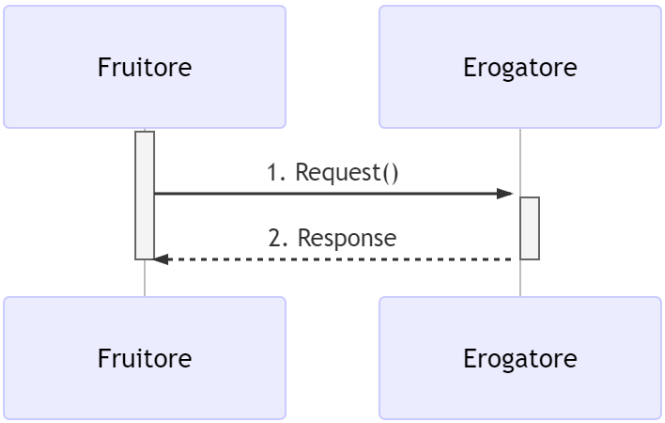


Figura 9 - Integrità del payload del messaggio

### 5.3.2 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il body del messaggio (ad esempio un oggetto JSON)
2. Il fruitore calcola il valore del Digest header dei representation data secondo le indicazioni in RFC 3230
3. Il fruitore individua l'elenco degli HTTP Header da firmare, incluso Digest e se presenti :httpheader:`Content-Type` e HTTP header Content-Encoding
4. Il fruitore crea la struttura o la stringa da firmare in modo che includa gli http header da proteggere, i riferimenti temporali di validità della firma e degli estremi della comunicazione, ovvero:
    i. il JOSE Header con almeno i parameter:
        a. alg con l'algoritmo di firma, vedi RFC 8725
        b. typ uguale a JWT

c. kid uguale all'identificativo della chiave pubblica, registrata su Piattaforma Digitale Nazionale Dati per l'interoperabilità, associata alla chiave privata utilizzata per la firma della request

ii. i seguenti claim obbligatori:

a. i riferimenti temporali di emissione e scadenza: iat , exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf.

b. il riferimento all'e-service dell'erogatore in aud;

iii. i seguenti claim, secondo la logica del servizio:

a. sub: oggetto (principal see :rfc:`3744#section-2`) dei claim contenuti nel jwt

b. iss: l'id del client utilizzato dal fruitore

c. jti: identificativo del JWT, per evitare replay attack

iv. il claim signed_headers con gli header http da proteggere ed i rispettivi valori, ovvero:

a. :httpheader:`Digest`

b. :httpheader:`Content-Type`

c. :httpheader:`Content-Encoding`

5. il fruitore firma il token adottando la JWS Compact Serialization
6. il fruitore posiziona il JWS nell'header Agid-JWT-Signature
7. Il fruitore spedisce il messaggio all'erogatore.

**B: Risultato**

8. L'erogatore decodifica il JWS presente in Agid-JWT-Signature header secondo le indicazioni contenute in :rfc:`7515#section-5.2`, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel Jose Header, in particolare verifica:

v. il contenuto dei claim iat , exp;

vi. la corrispondenza tra se stesso e il claim aud;

vii. l'univocità del claim jti se presente.

9. L'erogatore recupera dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità la chiave pubblica indicata dal fruitore nel claim kid del JOSE Header
10. L'erogatore valida la firma verificando l'elemento Signature del JWS
11. L'erogatore verifica la corrispondenza tra i valori degli header passati nel messaggio e quelli presenti nel claim signed-header, Content-Type e Content-Encoding se presenti devono essere sempre firmati, come indicato nel punto A3
12. L'erogatore quindi verifica la corrispondenza tra Digest ed il payload body ricevuto
13. Se le azioni da 8 a 12 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato del servizio richiamato.

Note:

- Per gli algoritmi da utilizzare in alg e Digest si vedano le Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

### 5.3.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

Richiesta HTTP con Digest e representation metadata

```
POST https://api.erogatore.example/rest/service/v1/hello/echo/ HTTP/1.1
Accept: application/json
Agid-JWT-Signature: eyJhbGciOiJSUzI1NiIsInR5c.vz8...
Digest: SHA-256=cFfTOCesrWTLVzxn8fmHl4AcrUs40Lv5D275FmAZ96E=
Content-Type: application/json

{"testo": "Ciao mondo"}
```

Porzione JWS con campi protetti dalla firma

```
# *header*
{
  "alg": "RS256",
  "typ": "JWT",
  "kid": "199d08d2-9971-4979-a78d-e6f7a544f296"
}
# *payload*

{
  "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
  "iat": 1516239022,
  "nbf": 1516239022,
  "exp": 1516239024,
  "signed_headers": [
    {"digest": "SHA-256=cFfTOCesrWTLVzxn8fmHl4AcrUs40Lv5D275FmAZ96E="},
    {"content-type": "application/json"}
  ],
}
```

Capitolo 6

# Inoltro dati tracciati nel dominio del Fruitore

## 6.1 [AUDIT_REST_01] Inoltro dati tracciati nel dominio del Fruitore REST

Il presente pattern aggiunge alla comunicazione tra fruitore ed erogatore a livello di messaggio:

- la capacità del fruitore di inoltrare i dati tracciati nel proprio dominio richiesti dall'erogatore.

Si adottano le indicazioni riportate in RFC 7231. Considereremo sempre richieste e risposte complete, con i metodi standard definiti in RFC 7231#section-4.

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, tramite il materiale crittografico depositato applicando i profili di emissione dei voucher previsti dalla stessa.

La costituzione del trust tra fruitore ed erogatore PUÒ essere realizzata al di fuori della Piattaforma Digitale Nazionale Dati per l'interoperabilità, attraverso l'utilizzo di materiale crittografico basato su certificati X.509, solo nel caso in cui il fruitore non possa accreditarsi alla stessa e comunque entro 12 mesi dal superamento di tale impedimento l'erogatore e fruitore devono aggiornare le modalità di costituzione del trust assicurando lo stesso per il tramite della Piattaforma Digitale Nazionale Dati per l'interoperabilità.

### 6.1.1 Descrizione

Il presente pattern declina l'utilizzo di:

- JSON Web Token (JWT) definita dall' RFC 7519;
- JSON Web Signature (JWS) definita dall' RFC 7515.

L'erogatore e il fruitore DEVONO concordare i dati tracciati dal fruitore nel proprio dominio richiesti dall'erogatore, individuando i claim da includere nel JWT di audit che, nel caso di utilizzo Piattaforma Digitale Nazionale Dati interoperabilità per la costruzione del trust, DEVONO essere debitamente descritti dall'erogatore nella documentazione allegata al relativo e-service pubblicato nel Catalogo API della Piattaforma Digitale Nazionale Dati interoperabilità.

Esempi di claim che POSSONO essere inclusi nel JWT di audit sono:

- userID, un identificativo univoco dell'utente interno al dominio del fruitore che ha determinato l'esigenza della request di accesso all'e-service dell'erogatore;
- userLocation, un identificativo univoco della postazione interna al dominio del fruitore da cui è avviata l'esigenza della request di accesso all'e-service dell'erogatore;
- LoA, livello di sicurezza o di garanzia adottato nel processo di autenticazione informatica nel dominio del fruitore.

Il fruitore DEVE sempre assicurare il popolamento dei seguenti claim del JWT di audit:

- "aud" il riferimento all'e-service dell'erogatore;

e, nel caso di utilizzo Piattaforma Digitale Nazionale Dati interoperabilità per la costruzione del trust, DEVE assicurare il popolamento del seguente claim del JWT di audit:

- "iss" il riferimento del client del fruitore utilizzato per la richiesta dell'e-service;
- "purposeId" l'id della finalità registrata dal fruitore sulla Piattaforma Digitale Nazionale Dati interoperabilità in relazione alla richiesta di fruizione dell'e-service.

Di seguito è descritta l'applicazione del presente pattern nei due scenari in cui il trust tra fruitore ed erogato è realizzato:

- per il tramite della Piattaforma Digitale Nazionale Dati interoperabilità (TRUST GESTITO DA PDND);
- al di fuori della Piattaforma Digitale Nazionale Dati interoperabilità (TRUST DIRETTO FRUITORE - EROGATORE).

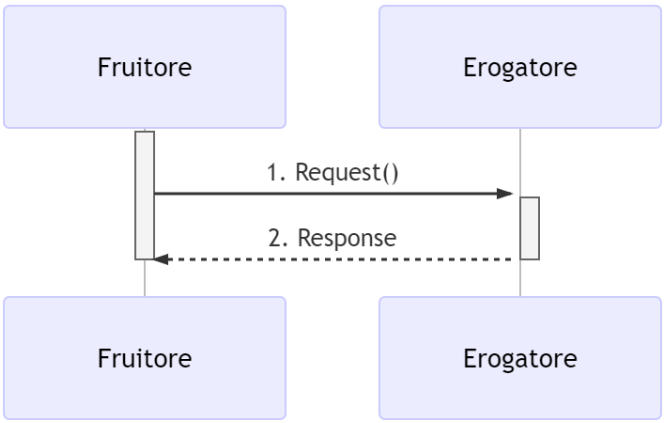


Figura 9 - Integrità del payload del messaggio

### 6.1.2 TRUST GESTITO DA PDND

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, nello specifico utilizzando i profili di emissione dei Voucher previsti per la Piattaforma Digitale Nazionale Dati per l'interoperabilità.

Per assicurare l'inoltro dei dati tracciati nel dominio del fruitore all'erogatore:

- il fruitore DEVE predisporre la rappresentazione dei dati tracciati e firmare la stessa utilizzando la chiave privata associata alla chiave pubblica registrata sulla Piattaforma Digitale Nazionale Dati per l'interoperabilità per il client utilizzato (JWS di audit);
- il fruitore nella request all'erogatore DEVE includere nell'header Agid-JWT-TrackingEvidence la rappresentazione dei dati tracciati e firmati (JWS di audit);
- l'erogatore DEVE verificare la firma del JWS di audit ricevuto nell'header Agid-JWT-TrackingEvidence, utilizzando la chiave pubblica recuperata dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità associata alla chiave privata utilizzata dal fruitore per la firma del JWS di audit.

In merito all'identificativo della chiave pubblica (kid) associata alla chiave privata utilizzata dal client, lo stesso è generato dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità.

Nell'attuazione dei precedenti passi il fruitore è responsabile della valorizzazione dei claim inclusi nel JWS di audit.

#### 6.1.2.1 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il JWS con i dati tracciati nel proprio dominio, ovvero:
    a. il JOSE Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725;
        ii. typ uguale a JWT;
        iii. kid uguale all'identificativo della chiave pubblica, registrata su Piattaforma Digitale Nazionale Dati per l'interoperabilità, associata alla chiave privata utilizzata per la firma;
    b. i seguenti claim obbligatori:
        i. i riferimenti temporali di emissione e scadenza: iat , exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf;
        ii. il riferimento dell'erogatore in aud;
        iii. l'id della finalità registrata dal fruitore su Piattaforma Digitale Nazionale Dati interoperabilità in purposeId;
        iv. l'id del client utilizzato dal fruitore in iss;
        v. identificativo del JWS in jti;
    c. i claim concordati con l'erogatore;
2. il fruitore firma il token adottando la JWS Compact Serialization utilizzando la chiave privata associata alla chiave pubblica registrata sulla Piattaforma Digitale Nazionale Dati per l'interoperabilità per il client utilizzato per la richiesta;
3. il fruitore posiziona il JWS di audit nell'header Agid-JWT-TrackingEvidence;
4. il fruitore spedisce il messaggio all'erogatore.

**B: Risultato**

5. L'erogatore decodifica il JWS di audit presente in Agid-JWT-TrackingEvidence header secondo le indicazioni contenute in :rfc:`7515#section-5.2`, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel Jose Header, in particolare verifica:
    a. il contenuto dei claim iat , exp;
    b. la corrispondenza tra se stesso e il claim aud;
6. l'erogatore recupera la chiave pubblica del client del fruitore dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità e valida la firma verificando il JWS di audit;
7. se l'azioni 5 e 6 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato dell'e-service richiamato.

Note:

- I precedenti passi 1 e 2 sono realizzati dal fruitore nella solo nel caso in cui non disponga di un digest del JWS di audit ancora valido nel proprio dominio;
- per gli algoritmi da utilizzare in alg e Digest si vedano le Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

#### 6.1.2.2 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

Richiesta HTTP

```
POST https://api.erogatore.example/rest/service/v1/hello/echo/ HTTP/1.1
Accept: application/json
Agid-JWT-TrackingEvidence: eyJhbGciOiJSUzI1NiIsInR5c.vz8...
Content-Type: application/json

{"testo": "Ciao mondo"}
```

Porzione JWS con campi protetti dalla firma

```
# *header*
{
  "alg": "ES256",
  "typ": "JWT",
  "kid": "199d08d2-9971-4979-a78d-e6f7a544f296"
}
# *payload*

{
  "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
  "iss": "be54418b-fa38-4060-bf11-eac2cc1a48ca",
  "purposeId": "4a153b51-5d47-4db9-be7e-e73dbcae4bb9",
  "iat": 1516239022,
  "nbf": 1516239022,
  "exp": 1516239024,
  "userID": "user293",
  "userLocation": "station012"
}
```

### 6.1.3 TRUST DIRETTO FRUITORE - EROGATORE

L'erogatore e il fruitore DEVONO definire il trust per consentire lo scambio del materiale crittografico necessario per assicurare la firma del JWS di audit.

Per dare seguito all'inoltro dei dati tracciati nel dominio del fruitore all'erogatore:

- il fruitore DEVE predisporre la rappresentazione dei dati tracciati e firmare la stessa utilizzando il materiale crittografico scambiato nel trust definito (JWS di audit), ove non disponga di una rappresentazione opaca dei dati tracciati e firmati già predisposta nei modi indicati ancora valida nel proprio dominio;
- il fruitore nella request all'erogatore deve includere nell'header Agid-JWT-TrackingEvidence la rappresentazione dei dati tracciati e firmati (JWS di audit);
- l'erogatore DEVE verificare la firma del JWS di audit ricevuto nell'header Agid-JWT-TrackingEvidence, utilizzando il materiale crittografico scambiato nel trust definito.

Nell'attuazione dei precedenti passi il fruitore è responsabile della valorizzazione dei claim inclusi nel JWS di audit.

#### 6.1.3.1 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il JWS con i dati tracciati nel proprio dominio, ovvero:
    a. il JOSE Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725
        ii. typ uguale a JWT
        iii. una o più delle seguenti opzioni per referenziare il certificato X.509:
            1. x5u (X.509 URL)
            2. x5c (X.509 Certificate Chain)
            3. x5t#S256 (X.509 Certificate SHA-256 Thumbprint
    b. i seguenti claim obbligatori:
        i. i riferimenti temporali di emissione e scadenza: iat , exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf.
        ii. il riferimento dell'erogatore in aud;
        iii. identificativo del JWS in jti;
    c. i claim concordati con l'erogatore;
2. il fruitore firma il token adottando la JWS Compact Serialization utilizzando il materiale crittografico scambiato nel trust definito;
3. il fruitore posiziona il JWS di audit nell'header Agid-JWT-TrackingEvidence;
4. il fruitore spedisce il messaggio all'erogatore.

**A: Richiesta**

5. L'erogatore decodifica il JWS di audit presente in Agid-JWT-TrackingEvidence header secondo le indicazioni contenute in :rfc:`7515#section-5.2`, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel Jose Header, in particolare verifica:
    a. il contenuto dei claim iat , exp;
    b. la corrispondenza tra se stesso e il claim aud;

6. l'erogatore recupera il certificato X.509 referenziato nel JOSE Header facendo attenzione alle indicazioni contenute in :rfc:`8725#section-3.10`;
7. l'erogatore verifica il certificato secondo i criteri del trust;
8. l'erogatore valida la firma verificando il JWS di audit con il materiale crittografico scambiato nel trust definito;
9. se l'azioni da 5 a 8 hanno avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato dell'e-service richiamato.

Note:

- I precedenti passi 1 e 2 sono realizzati dal fruitore nella solo nel caso in cui non disponga di un digest del JWS di audit ancora valido nel proprio dominio;
- I precedenti passi 1 e 2 sono realizzati dal fruitore nella solo nel caso in cui non disponga di un digest del JWS di audit ancora valido nel proprio dominio;

#### 6.1.3.2 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore.

Richiesta HTTP

```
POST https://api.erogatore.example/rest/service/v1/hello/echo/ HTTP/1.1
Accept: application/json
Agid-JWT-TrackingEvidence: eyJhbGciOiJSUzI1NiIsInR5c.vz8...
Content-Type: application/json

{"testo": "Ciao mondo"}
```

Porzione JWS con campi protetti dalla firma

```
# *header*
 {
   "alg": "ES256",
   "typ": "JWT",
   "x5c": [
      "MIICyzCCAbOgAwIBAgIEC..."
]
 }
 # *payload*
```

```
{
  "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
  "iat": 1516239022,
  "nbf": 1516239022,
  "exp": 1516239024,
  "userID": "user293",
  "userLocation": "station012"
}
```

## 6.2 [AUDIT_REST_02] Inoltro dati tracciati nel dominio del Fruitore REST con correlazione

Il presente pattern aggiunge alla comunicazione tra fruitore ed erogatore a livello di messaggio:

- la capacità del fruitore di inoltrare i dati tracciati nel proprio dominio richiesti dall'erogatore;
- la correlazione tra gli strumenti di autenticazione e i dati tracciati nel proprio dominio e inoltrati dal fruitore.

Si adottano le indicazioni riportate in RFC 7231. Considereremo sempre richieste e risposte complete, con i metodi standard definiti in RFC 7231#section-4.

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, tramite il materiale crittografico depositato applicando i profili di emissione dei voucher previsti dalla stessa.

### 6.2.1 Descrizione

Il presente pattern declina l'utilizzo di:

- JSON Web Token (JWT) definita dall' RFC 7519;
- JSON Web Signature (JWS) definita dall' RFC 7515.

L'erogatore e il fruitore DEVONO concordare i dati tracciati dal fruitore nel proprio dominio richiesti dall'erogatore, individuando i claim da includere nel JWT di audit che, nel caso

di utilizzo Piattaforma Digitale Nazionale Dati interoperabilità per la costruzione del trust, DEVONO essere debitamente descritti dall'erogatore nella documentazione allegata al relativo e-service pubblicato nel Catalogo API della Piattaforma Digitale Nazionale Dati interoperabilità.

Esempi di claim che POSSONO essere inclusi nel JWT di audit sono:

- userID, un identificativo univoco dell'utente interno al dominio del fruitore che ha determinato l'esigenza della request di accesso all'e-service dell'erogatore;
- userLocation, un identificativo univoco della postazione interna al dominio del fruitore da cui è avviata l'esigenza della request di accesso all'e-service dell'erogatore;
- LoA, livello di sicurezza o di garanzia adottato nel processo di autenticazione informatica nel dominio del fruitore.

Il fruitore DEVE sempre assicurare il popolamento dei seguenti claim del JWT di audit:

- "dnonce" un numero casuale costituito da 13 cifre, al fine di aumentare l'entropia dello stesso;
- "aud" il riferimento all'e-service dell'erogatore;
- "iss" il riferimento del fruitore o del client utilizzato per la richiesta dell'e-service;
- "purposeId" l'id della finalità registrata dal fruitore sulla Piattaforma Digitale Nazionale Dati interoperabilità in relazione alla richiesta di fruizione dell'e-service.

L'erogatore e il fruitore DEVONO utilizzare la Piattaforma Digitale Nazionale Dati per l'interoperabilità di cui al comma 2 dell'articolo 50-ter del CAD per la costituzione del trust, nello specifico ai profili di emissione dei Voucher previsti per la Piattaforma Digitale Nazionale Dati per l'interoperabilità sono aggiunti i seguenti passi per garantire la non ripudiabilità del contenuto del JWT di audit:

- il fruitore, applicando quanto indicato nelle specifiche tecniche della Piattaforma Digitale Nazionale Dati per l'interoperabilità, DEVE predisporre la rappresentazione opaca dei dati tracciati e firmati (digest del JWS di audit), o utilizzare una rappresentazione opaca dei dati tracciati e firmati ancora valida nel

proprio dominio, ed inserirla nella Access Token Request alla Piattaforma Digitale Nazionale Dati per l'interoperabilità;

- la Piattaforma Digitale Nazionale Dati per l'interoperabilità DEVE inserire la rappresentazione opaca dei dati tracciati(digest del JWS di audit) nell'Access Token, ovvero il Voucher rilasciato al fruitore;
- il fruitore nella request all'erogatore deve includere nell'header Agid-JWT-TrackingEvidence la rappresentazione dei dati tracciati e firmati (JWS di audit);
- l'erogatore DEVE verificare la firma del JWS di audit ricevuto nell'header Agid-JWT-TrackingEvidence, utilizzando la chiave pubblica recuperata dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità;
- l'erogatore DEVE calcolare il digest della rappresentazione dei dati tracciati e firmati (JWS di Audit) ricevuti nell'header Agid-JWT-TrackingEvidence e verificarne la corrispondenza con quanto presente nell'Access Token (digest JWS di Audit).

Nell'attuazione dei precedenti passi il fruitore è responsabile della:

- valorizzazione dei claim inclusi nel JWS di audit;
- opacizzazione dei dati tracciati inoltrata alla Piattaforma Digitale Nazionale Dati per l'interoperabilità.

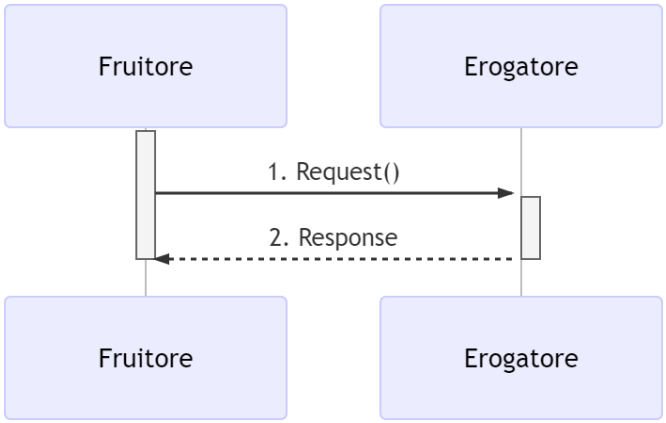


Figura 9 - Integrità del payload del messaggio

### 6.2.2 Regole di processamento

La creazione ed il processamento dei JWT DEVE rispettare le buone prassi di sicurezza indicate in RFC 8725.

**A: Richiesta**

1. Il fruitore predispone il JWS con i dati tracciati nel proprio dominio, ovvero:
    a. il JOSE Header con almeno i parameter:
        i. alg con l'algoritmo di firma, vedi RFC 8725;
        ii. typ uguale a JWT;
        iii. kid uguale all'identificativo della chiave pubblica, registrata su Piattaforma Digitale Nazionale Dati per l'interoperabilità, associata alla chiave privata utilizzata per la firma della request;
    b. i seguenti claim obbligatori:
        i. i riferimenti temporali di emissione e scadenza: iat , exp. Se il flusso richiede di verificare l'istante di prima validità del token, si può usare il claim nbf;
        ii. il riferimento dell'erogatore in aud;
        iii. l'id della finalità registrata dal fruitore su Piattaforma Digitale Nazionale Dati interoperabilità in purposeId;
        iv. l'id del client utilizzato dal fruitore in iss;
        v. identificativo del JWS in jti;
        vi. id della finalità registrata dal fruitore in purposeId;
        vii. un numero casuale costituito da 13 cifre in dnonce;
    c. i claim concordati con l'erogatore;
2. il fruitore firma il token adottando la JWS Compact Serialization utilizzando la chiave privata associata alla chiave pubblica registrata sulla Piattaforma Digitale Nazionale Dati per l'interoperabilità al client utilizzato per la richiesta;
3. il fruitore calcola il digest del JWS di audit e lo aggiunge alla richiesta del Voucher secondo le modalità indicate nelle specifiche tecniche della Piattaforma Digitale Nazionale Dati per l'interoperabilità;
4. il fruitore posiziona il Voucher nell'header Autorization e il JWS di audit nell'header Agid-JWT-TrackingEvidence;

5. il fruitore spedisce il messaggio all'erogatore.

**B: Risultato**

6. L'erogatore verifica il Voucher secondo le modalità indicate nelle specifiche tecniche della Piattaforma Digitale Nazionale Dati per l'interoperabilità.
7. L'erogatore decodifica il JWS di audit presente in Agid-JWT-TrackingEvidence header secondo le indicazioni contenute in :rfc:`7515#section-5.2`, le buone prassi indicate in RFC 8725 e valida i claim contenuti nel Jose Header, in particolare verifica:
    a. il contenuto dei claim iat , exp;
    b. la corrispondenza tra se stesso e il claim aud;
8. l'erogatore verifica la corrispondenza del digest contenuto nel Voucher della Piattaforma Digitale Nazionale Dati per l'interoperabilità è il digest calcolato dal JWS di audit presente nell'header Agid-JWT-TrackingEvidence;
9. l'erogatore recupera la chiave pubblica del client del fruitore dalla Piattaforma Digitale Nazionale Dati per l'interoperabilità e valida la firma verificando l'elemento Signature del JWS di audit;
10. se l'azioni 6 o 9 ha avuto esito positivo, il messaggio viene elaborato e viene restituito il risultato dell'e-service richiamato.

Note:

- I precedenti passi 1, 2 e 3 sono realizzati dal fruitore nella solo nel caso in cui non disponga di un digest del JWS di audit ancora valido nel proprio dominio;
- per gli algoritmi da utilizzare in alg e Digest si vedano le Linee Guida sulla sicurezza, emanate dall'Agenzia per l'Italia Digitale ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 (Codice dell'Amministrazione Digitale).

### 6.2.3 Esempio

Di seguito è riportato un tracciato del messaggio inoltrato dal fruitore all'interfaccia di servizio dell'erogatore. Richiesta HTTP con Digest e representation metadata

Richiesta HTTP

```
POST https://api.erogatore.example/rest/service/v1/hello/echo/ HTTP/1.1
Accept: application/json
Autorization: Bearer AftgSSDGciFEEOiJfsI1Nfsdfsdfi IsInR5c.vfd5...
Agid-JWT-TrackingEvidence: eyJhbGciOiJSUzI1NiIsInR5c.vz8...
Digest: SHA-256=cFfTOCesrWTLVzxn8fmHl4AcrUs40Lv5D275FmAZ96E=
Content-Type: application/json

{"testo": "Ciao mondo"}
```

Porzione JWS con campi protetti dalla firma

```
# *header*
{
  "alg": "ES256",
  "typ": "JWT",
  "kid": "199d08d2-9971-4979-a78d-e6f7a544f296"
}
# *payload*

{
  "aud": "https://api.erogatore.example/rest/service/v1/hello/echo"
  "iss": "be54418b-fa38-4060-bf11-eac2cc1a48ca",
  "purposeId": "4a153b51-5d47-4db9-be7e-e73dbcae4bb9",
  "dnonce": 1234567890123,
  "iat": 1516239022,
  "nbf": 1516239022,
  "exp": 1516239024,
  "userID": "user293",
  "userLocation": "station012"
}
```