



AGID

Agenzia per l'Italia Digitale

spod

Token R.A.O.Pubblico

Pag. 1/19

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



Presidenza del Consiglio dei Ministri



Definizione formato *Token* per R.A.O. pubblico

Definizione formato *token* per R.A.O. pubblico

DATA: 21/09/2020

VERSIONE - STATO: 1.2

AUTORI: AGENZIA PER L'ITALIA DIGITALE

PROGETTO: SPID

Version History

VERSIONE	DATA	STATO	MODIFICHE
1.0	21/11/2019	PUBBLICATO	PRIMA STESURA
1.1	15/07/2020	REVISIONE	REVISIONE ESEMPI
1.2	21/09/2020	REVISIONE	REVISIONE CODIFICHE





Sommario

Definizione formato <i>Token</i> per R.A.O. pubblico	2
Version History	2
1. Obiettivi.....	4
2. ICRequestData.....	4
3. <i>Token Completo</i>	7
4. Funzione di cifratura	9
5. Sigillo Elettronico	9
6. Modello interscambio dati – modello a).....	9
7. API – modello a)	10
8. Upload del <i>Token Completo</i> – modello b).....	11
9. Verifiche di validità del <i>Token Completo</i>	11
10. Appendice	13
10.1 Json Schema per ICRequestData.....	13
10.2 Json Schema per <i>token completo</i>	17
10.3 Json Schema per Response Payload	18
10.4 Riferimenti PKI AgID	19





1. Obiettivi

Definizione di uno standard condiviso, chiaro e sicuro per il formato dei dati utili al rilascio di una identità SPID a fronte di un avvenuto riconoscimento del futuro titolare dell'identità presso uffici pubblici.

2. ICRequestData

L'ICRequestData (Identity Creation Request Data) è l'oggetto contenente i dati relativi all'utente che ha eseguito la sua identificazione tramite R.A.O. pubblico e necessari alla generazione di una identità digitale SPID presso uno degli Identity Provider SPID che aderiscono al Modello R.A.O. pubblico.

Tali dati sono rappresentati tramite un documento definito nel formato JSON (JavaScript Object Notation), il cui schema è riportato nell'Appendice 10.1 Json Schema per ICRequestData e di cui un esempio è riportato nell'Esempio 1.

In particolare, l'ICRequestData riporta i seguenti dati:

- **info**: elemento che contiene le seguenti informazioni:
 - **id**: codice identificativo unico della richiesta all'interno di uno specifico R.A.O. pubblico.
 - **issueInstant**: istante di generazione della richiesta, codificato come NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT)¹.
 - **issuer**: elemento che consente l'identificazione del R.A.O. pubblico emittente le richieste, codificato come:
 - **issuerCode**: P.A. agente come R.A.O. pubblico, identificata tramite il proprio codice IPA.
 - **issuerInternalReference** (*opzionale*): dato avente significato solo per il R.A.O. pubblico e da esso definito liberamente, utile per eventuali verifiche in caso di necessità. Esso non deve essere più lungo di 32 caratteri.
- **electronicIdentification**: elemento che contiene le seguenti informazioni del documento presentato per l'identificazione elettronica:
 - **identificationType**: indica la tipologia del documento. I valori ammessi sono "TS" e "CF", rispettivamente relativi alla Tessera Sanitaria e al Tesserino del Codice Fiscale fornito ai residenti all'estero sprovvisti di Tessera Sanitaria.
 - **identificationSerialCode**: contiene il seriale della tessera utilizzata per l'identificazione.
 - **identificationExpirationDate**: riporta la data di fine validità indicata sulla tessera utilizzata per l'identificazione, codificata nel formato YYYY-mm-dd.

¹ <https://tools.ietf.org/html/rfc7519>





- **spidAttributes:** elemento che contiene i seguenti dati dell'utente identificato dal R.A.O. pubblico:
 - **mandatoryAttributes:** contiene i seguenti dati necessari per la generazione dell'identità digitale SPID per una persona fisica:
 - **name:** Nome della persona fisica, codificato come previsto dalla Tabella Attributi SPID
 - **familyName:** Cognome della persona fisica, codificato come previsto dalla Tabella Attributi SPID
 - **placeOfBirth:** Luogo di nascita della persona fisica, codificato come previsto dalla Tabella Attributi SPID
 - **countyOfBirth:** Provincia di nascita della persona fisica, codificata come previsto dalla Tabella Attributi SPID
 - **nationOfBirth:** Nazione di nascita della persona fisica, codificata con il codice catastale della nazione definito dall'ISTAT. Ad esempio, nel caso dell'Italia indicare Z000, se invece la nazione è sconosciuta indicare Z998.
 - **dateOfBirth:** Data di nascita della persona fisica, codificata come previsto dalla Tabella Attributi SPID.
 - **gender:** Sesso della persona fisica, codificata come previsto dalla Tabella Attributi SPID.
 - **fiscalNumber:** Codice Fiscale della persona fisica, codificata come previsto dalla Tabella Attributi SPID.
 - **email:** Indirizzo di posta elettronica, codificata come previsto dalla Tabella Attributi SPID.
 - **idCard:** Dati relativi al documento di identità fornito dalla persona fisica al momento della sua identificazione presso il R.A.O. pubblico, indicati come segue:
 - **idCardType:** Tipo del documento, codificato secondo i valori ammessi indicati nella Tabella Attributi SPID.
 - **idCardDocNumber:** Numero del documento.
 - **idCardIssuer:** Nome dell'ente emettitore del documento, codificata come previsto dalla Tabella Attributi SPID.
 - **idCardIssueDate:** data di rilascio del documento, codificata come previsto dalla Tabella Attributi SPID.
 - **idCardExpirationDate:** data di scadenza del documento, codificata come previsto dalla Tabella Attributi SPID.
 - **mobilePhone:** numero di cellulare, codificato come segue:
 - **countryCallingCode:** Prefisso internazionale dell'operatore, codificato secondo standard ITU (ad esempio, +39).
 - **phoneNumber:** numero di telefonia mobile, codificato come stringa numerica senza spazi intermedi.



- **address:** Domicilio fisico della persona fisica, codificato come segue:
 - **type:** Tipologia del luogo (via, viale, piazza ...);
 - **addressName:** Nome del luogo
 - **addressNumber:** Numero civico
 - **postalCode:** Codice Postale del luogo
 - **municipality:** Comune a cui è afferente il luogo, codificato tramite codice catastale (Codice Belfiore).
 - **county:** Provincia a cui è afferente il luogo, codificata tramite sigla.
 - **nation:** Nazione a cui è afferente il luogo, codificata con il codice catastale della nazione definito dall'ISTAT. Ad esempio, nel caso dell'Italia indicare Z000, se invece la nazione è sconosciuta indicare Z998.
- **optionalAttributes (opzionale):** ulteriore informazione che può essere inclusa all'interno dell'identità digitale SPID per una persona fisica:
 - **digitalAddress:** Domicilio Digitale.

Esempio 1: Esempio di ICRequestData

```
{
  "info": {
    "id": "123456789",
    "issueInstant": "1600696800",
    "issuer": {
      "issuerCode": "c_h501",
      "issuerInternalReference": "03Ab!34T"
    }
  },
  "electronicIdentification": {
    "identificationType": "TS",
    "identificationSerialCode": "123456789",
    "identificationExpirationDate": "2023-09-24"
  },
  "spidAttributes": {
    "mandatoryAttributes": {
      "name": "Giovanni Mario",
      "familyName": "Rossi",
      "placeOfBirth": "F205",
      "countyOfBirth": "MI",
      "nationOfBirth": "Z000",
      "dateOfBirth": "2000-09-24",
      "gender": "M",
      "fiscalNumber": "TINIT-RSSGNN00P24F205L",
      "email": "me@me.com",
      "idCard": {
        "idCardType": "cartalidentita",
        "idCardDocNumber": "AS09452389",
      }
    }
  }
}
```



```
"idCardIssuer": "c_h501",
"idCardIssueDate": "2013-01-02",
"idCardExpirationDate": "2023-09-24"
},
"mobilePhone": {
  "countryCallingCode": "+39",
  "phoneNumber": "3471234567"
},
"address": {
  "addressType": "Largo",
  "addressName": "Augusto",
  "addressNumber": "3/b",
  "postalCode": "00129",
  "municipality": "H501",
  "county": "RM",
  "nation": "Z000"
}
},
"optionalAttributes": {
  "digitalAddress": "me@meypecprovider.com"
}
}
}
```

3. Token Completo

Il *token completo* è formalizzato come un JWT (**JSON Web token**), generato a partire da un payload, definito di seguito e sigillato secondo le indicazioni del paragrafo 5. Sigillo Elettronico.

Il *token* ha validità di 30 giorni, periodo in cui l'utente, a cui fanno riferimento le informazioni contenute nel *token*, può utilizzarlo per ottenere un'identità digitale. Trascorso tale termine, il *token* non è più usabile e, nel caso previsto al punto a) del paragrafo 3.6 delle Linee Guida, l'Identity Provider provvede alla sua cancellazione.

L'header del *token* JWT è costituito dalle seguenti informazioni:

- **typ**: parametro valorizzato come "JWT".
- **alg**: parametro che identifica l'algoritmo crittografico del sigillo elettronico utilizzato.
- **x5c**: parametro contenente il certificato o la catena dei certificati, in formato X.509, corrispondente alla chiave pubblica del certificato di sigillo elettronico utilizzato. Il certificato o la catena dei certificati sono codificati come array JSON di stringhe corrispondenti ai valori dei certificati in formato DER. Ogni stringa nell'array è codificata in Base64. Il certificato contenente la chiave pubblica del sigillo utilizzato per firmare il *token*



deve essere la prima stringa dell'array. Il certificato di sigillo elettronico può essere lo stesso eventualmente utilizzato dal client per il protocollo di comunicazione HTTPS.

Il payload è rappresentato da un documento definito nel formato JSON (JavaScript Object Notation), il cui schema è riportato nell'Appendice 10.2 Json Schema per *token completo*, contenente le seguenti informazioni:

- **iss**: corrispondente al valore degli elementi *issuerCode* e *issuerInternalReference* dell'elemento *info* in *ICRequestData* codificati singolarmente in Base64 e concatenati tramite punto. Esempio: `Base64(issuerCode).Base64(issuerInternalReference)`
- **sub**: corrispondente al valore dell'elemento *id* dell'elemento *info* in *ICRequestData*
- **jti**: identificativo unico del *token*, generato come UUID.
- **aud**: tale attributo deve essere presente solo nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. a) delle Linee Guida e valorizzato con l'entityID dell'IdP come indicato nel registro SPID. Nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. b) delle Linee Guida, invece, l'attributo non deve essere presente all'interno del payload del token JWT.
- **iat**: istante di generazione della richiesta, codificato come *NumericDate* come indicato in RFC 7519 – JSON Web Token (JWT)¹. A partire da tale riferimento temporale vengono conteggiati i 30 giorni di validità del *token*. Deve corrispondere al valore dell'elemento *issueInstant* in *ICRequestData*.
- **exp**: tempo di fine validità del *token*, codificato come *NumericDate*, come indicato in RFC 7519 – JSON Web Token (JWT)¹, calcolato come `iat + 30` giorni.
- **fiscalNumber**: codice fiscale della persona fisica che ha eseguito la sua identificazione tramite R.A.O. pubblico.
- **encryptedData**: versione serializzata e cifrata dell'*ICRequestData*, secondo le specifiche indicate nel paragrafo **Errore. L'autoriferimento non è valido per un segnalibro..**

Esempio 2: Esempio di Payload JSON

valido per il modello di riferimento di cui al paragrafo 3.6, lett. b) delle Linee Guida
encryptedData codificata a partire dalla passphrase **#-MIK-Pass2#**

```
{
  "iss": "Y19oNTAx.MDNBYiEzNFQ=",
  "sub": "123456789",
  "jti": "822e653a-d504-420c-9da3-609b329fc6b5",
  "iat": "1600696800",
  "exp": "1603288800",
  "fiscalNumber": "RSSGNN00P24F205L",
  "encryptedData": "
eyJhbGciOiJkaXIiLCJlbmMiOiJBMjU2Q0JDLUhTNTUyLn0..HlJlqoSCZVLH4NmN9U1Fcw.wNwBHIVthYnLNfbzRJ7ZC
QvGgRFdRYC_SZCrLQK4z9MeJZ9xNEDUejvn4bpo5R-RkhvHeZr7QMpS6qaNYnrXpzDVR89xF--
6ueNrzDowXmLu2jl-9nl4V7jGGR5F7ebtU_M3meTaCz-ku5vo5ipNsk-OINjt13mblho5Z1IEAcYA_238VeDO-
```





```
BA0WjBaaZEhQW5it9Mi2TQ1qZnjgL2SzKxppqJaaAUCRbUUKnyNmXgnoQI2fz-  
rQuugwQRJYOGnrTbeaBieGkp9p6biQGG2my250neB0iEWx6s1o5WJm8Mw8WFGHseK3rNUWho38gayPnhZvZg  
MHUiiWCEgg10hm8pcbT_VShNvhtunrNXZwUhx53o3licb4oyaPsgl3t-  
7kJwrOyQfkekFZXX18fzqi0UNSlzL2gwoBUhivymReBUuL3oN0Qqg2e4cYclg4H2Dn38nwck8TycX-  
u3w33GMaPXKDDsNLTTBEuVslwNDKYZNIIGFwXawjv2WYewiY-  
F8SOXN11ixlbXIAMMdpzj_Ep7YdeFO9WSMy5YHJ69C33uqd8i7kg9HE5LwLgCZn7zErYDgJ2VpzGChm8gArtEary  
wVK1Xq_mCwlbO4On41eUqsej9Nlq9vG0v-  
0mlU0eLwEhcOp6Z39vDIKPbcgSwrrxFmR0GQbf2MX4Mwf9BSaUgdf0ajXopr_tunlhMJrPvVvaMCQQ6Da-  
Hkl0WBRR5i36bHAPkRVAgT1JjI3M_22I7XyRhBl0rhpzo53BV3i3EyB7CPF3pLT1qF-  
teUGZc6QgWuFzf5I1n_b9AP9KZ1QHz3qayn4_eJ709M-4RqUfvWIC5ITiw16oTb_VC-  
uQ_uugypFAGq8mFRI7VnYQ9nqhcvPdUl79o_BvdKNYQsLhYsf6RABdnOI5rP7r7w39ZDtN2ZEEyD65otRjcszG4405J  
jyCeJahV8BnGMZ901S9ziWgH_6rNEJxck6KjVT47nRxS5vqQIFMMnjGfEkg2RxsFlug_9OmrquwNMSHvoNZI6uXVT  
jNVuEmf4f9luN5Jx_Zf81tkFbhsTN-3WOPjsHz7Tb4RoOK4mdwYdGKYRFSErCx03nTcy3N-  
U1EdJY5vtDtABubrv8p3s1dtblGc8sYgCV66QtumTqLIG29UmTPTU2ApEPH06cUy7YSj0LSD0hR86KkKektfyKjiTkOM  
07y_gtwR3hT6mr-  
vof4iKrbOFQ28Ked9GcF3UWYxh8pSmGAhGHuljsvS_S_rEiKyOMRHkK66B8DB1zwsI6v_ZZxtZgXVcWH6wrnBxll_6  
D8A.OWjas4sGqp-802eXr85LDE1ILVY6c_FyyAF5gDYy1QU"  
}
```

4. Funzione di cifratura

Per la generazione dell'EncryptedData a partire dell'ICRequestData è utilizzato lo standard JWE (JSON Web Encryption).

In particolare, per la cifratura del payload JSON, che sarà costituito dalla struttura ICRequestData deve essere utilizzata la modalità di direct encryption con algoritmo di cifratura A256CBC-HS512².

La chiave di cifratura utilizzata dovrà essere di 512 bit (64 byte), generata tramite funzione di hash crittografica SHA-512 a partire dalla passphrase fornita dall'utente in fase di richiesta di generazione della propria identità digitale.

5. Sigillo Elettronico

Il *token completo* è oggetto di un sigillo elettronico, basato su un certificato emesso da apposita sub CA generata dall'Agenzia.

6. Modello interscambio dati – modello a)

Nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. a) delle Linee Guida, al fine di consentire la comunicazione, da parte dei RAO verso gli IdP, del *token* definito nei paragrafi

² Rif. <https://tools.ietf.org/html/rfc7518>





precedenti e la comunicazione, da parte degli IdP verso i RAO, del risultato dell'invio del predetto *token*, si definisce un modello di interscambio dati che prevede:

- Esposizione da parte dell'IdP di un opportuno Endpoint, espresso tramite URL HTTPS, avente i seguenti requisiti:
 - 1) **Algoritmo di cifratura:** il canale supporta esclusivamente gli algoritmi TLS 1.2 e/o TLS 1.3. Tutti gli altri algoritmi (ad es. SSL3, TLS 1.0) non sono supportati.
 - 2) **Cipher Suites:** il canale non supporta suite di cifratura anonime.
- Comunicazione fra R.A.O. pubblico ed IdP svolta attraverso API REST esposta dall'IdP (Vedi 7. API – modello a)7. API) il cui accesso dovrà essere limitato ai R.A.O. pubblici, tramite la verifica dell'uso dei rispettivi sigilli del R.A.O. pubblico e dell'IdP per instaurare il canale TLS.

7. API – modello a)

In conformità con il modello di cui al paragrafo 6. Modello interscambio dati – modello a), gli IdP dovranno esporre un endpoint denominato */raoic* (Registration Authority Office Identity Creation). La url completa dell'endpoint, esposta su dominio appartenente all'IdP, dovrà essere comunicato all'Agenzia e da quest'ultima pubblicata su apposito registro.

L'endpoint potrà ricevere solo richieste di tipo POST contenenti nel body il *token* JWT come indicato nel paragrafo

3. Token *Completo*. Ogni altro tipo di richiesta inviata tramite diverso binding causerà un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico).

A seguito della ricezione di un *token completo*, l'IdP effettua le verifiche di cui al nel paragrafo 9. Verifiche di validità del *Token Completo*. Al completamento della verifica l'IdP genererà una response coerente con l'esito della verifica stessa.

Ogni response sarà restituita come oggetto JSON in formato JWT firmato indicante l'esito dell'invio e/o la causa del diniego. L'header del *token* JWT della response è costituito dalle seguenti informazioni:

- **typ:** parametro valorizzato come "JWT";
- **alg:** parametro che identifica l'algoritmo crittografico del sigillo elettronico utilizzato;
- **x5c:** parametro contenente il certificato o la catena dei certificati, in formato X.509, corrispondente alla chiave pubblica del certificato di sigillo elettronico dell'IdP. Il certificato o la catena dei certificati sono codificati come array JSON di stringhe corrispondenti ai valori dei certificati in formato DER. Ogni stringa nell'array è codificata in Base64. Il certificato contenente la chiave pubblica del sigillo utilizzato per firmare il *token* deve essere la prima



stringa dell'array. Il certificato di sigillo elettronico può essere lo stesso eventualmente utilizzato dal client per il protocollo di comunicazione HTTPS.

Il payload della response è rappresentato da un documento definito nel formato JSON (JavaScript Object Notation), definito secondo lo schema riportato nell'Appendice 10.3 Json Schema per Response Payload:

- **iss**: corrispondente all'entityID dell'IdP, come indicato nel registro SPID;
- **sub**: corrispondente al valore dell'elemento *id* dell'elemento *info* in ICRequestData;
- **jti**: identificativo unico del *token*, generato come UUID;
- **aud**: corrispondente al valore *iss* contenuto nella request;
- **iat**: istante di generazione della response, codificato come timestamp Unix;
- **responseCode**: codice dell'esito;
- **responseMessage**: messaggio relativo all'esito

Esempio 3: Esempio di Payload del JWT della Response

```
{
  "iss": "www.idp.it",
  "sub": "123456789",
  "jti": "a7388c12-ea4a-43fe-b5ad-befd4a9edf81",
  "aud": " Y19oNTAx.MDNBYiEzNFQ=",
  "iat": "2019-06-27T15:55:03.405Z",
  "responseCode": 2,
  "responseMessage": "Spiacenti, per questo utente risulta già rilasciata un'identità digitale SPID"
}
```

8. Upload del Token Completo- modello b)

Nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. b) delle Linee Guida, gli IdP forniranno all'utente la possibilità presso il loro sito di selezionare la modalità di rilascio con "identificazione tramite P.A." e procedere all'upload del *token completo*.

A seguito della ricezione di un *token completo*, l'IdP effettua le verifiche di cui al nel paragrafo 9. Verifiche di validità del *Token Completo*. Al completamento della verifica l'IdP notificherà all'utente un messaggio coerente con l'esito della verifica stessa.

9. Verifiche di validità del *Token Completo*

Alla ricezione del *token* l'IdP dovrà verificare che:



1. Il *token* ricevuto è conforme a quanto previsto dal paragrafo 3. *Token Completo*, in caso contrario verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico);
2. Il valore dell'algoritmo di firma indicato nel campo *alg* sia tra quelli previsti per i certificati emessi da sub CA dell'Agenzia. In caso contrario verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico);
3. La firma sul token sia valida e che il certificato indicato nel campo *x5c* sia rilasciato da PKI dell'Agenzia e sia valido e non revocato. In caso contrario verrà generato un errore di tipo Unauthorized (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico). Per la verifica dei certificati occorre utilizzare i certificati di Root CA e Subordinate CA per Enti pubblici e la Certificate Revocation List, i cui riferimenti sono indicati all'allegato 10.4 *Riferimenti PKI AgID*;
4. Solo nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. a) delle Linee Guida, il valore indicato nel campo *aud* del body corrisponda al proprio entityID. In caso contrario verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico);
5. Solo nel caso previsto dal modello di riferimento di cui al paragrafo 3.6, lett. a) delle Linee Guida, il valore indicato nel campo *iat* rientri in un intervallo di 10 minuti nell'intervallo dell'istante corrente ($istante\ attuale - 5min < iat < istante\ attuale + 5min$). In caso contrario verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico);
6. Il valore indicato nel campo *exp* corrisponda a $iat + 30$ giorni. In caso contrario verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico);
7. Il valore indicato nel campo *exp* sia maggiore dell'istante corrente. In caso contrario verrà generato un errore di tipo Expired Token (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico).

Effettuate tali verifiche l'IdP potrà decodificare il valore del campo *encryptedData* e verificare che i valori per i campi *id*, *issuInstant* e *issuer* corrispondano rispettivamente a *sub*, *iat*, *aud*, come da specifiche nel paragrafo 3. *Token Completo*.

In caso negativo verrà generato un errore di tipo Bad Request (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico). Altrimenti i dati potranno essere salvati e verrà generato un evento di tipo Ok (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico).

Nel caso in cui l'identità sia già presente presso l'IdP, verrà generato un errore di tipo User Exists (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico).

Nel caso in cui sia già presente un *token* valido per l'identità presso l'IdP, verrà generato un evento di tipo *Token Exists* (vedi l'allegato Tabella messaggi *token* R.A.O. pubblico) e l'IdP procederà a sovrascrivere il vecchio *token* con il nuovo.



10. Appendice

10.1 Json Schema per ICRequestData

Il Json Schema utile per la validazione dei ICRequestData è illustrato nella tabella A.1

Tabella A.1

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "ICRequestData Schema",
  "description": "",
  "type": "object",
  "properties": {
    "Info": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "issueInstant": {
          "type": "integer"
        },
        "issuer": {
          "type": "object",
          "properties": {
            "issuerCode": {
              "type": "string"
            },
            "issuerInternalReference": {
              "type": "string",
              "maxLength": 32
            }
          }
        },
        "required": [
          "issuerCode",
          "issuerOfficeCode"
        ]
      }
    },
    "required": [
      "ICRequestID",
      "ICRequestIstant",
      "ICRequestIssuer"
    ],
    "electronicIdentification": {
      "type": "object",
      "properties": {
```



```
"identificationType": {
  "enum": [
    "TS",
    "CF"
  ]
},
"identificationSerialCode": {
  "type": "string"
}
},
"required": [
  "identificationType",
  "identificationSerialCode"
]
},
"spidAttributes": {
  "type": "object",
  "properties": {
    "mandatoryAttributes": {
      "type": "object",
      "properties": {
        "name": {
          "type": "string"
        },
        "familyName": {
          "type": "string"
        },
        "placeOfBirth": {
          "type": "string",
          "pattern": "[A-Z][0-9]{3}"
        },
        "countyOfBirth": {
          "type": "string",
          "maxLength": 2
        },
        "nationOfBirth": {
          "type": "string",
          "pattern": "Z[0-9]{3}"
        },
        "dateOfBirth": {
          "type": "string",
          "format": "date"
        },
        "gender": {
          "enum": [
            "M",
            "F"
          ]
        }
      }
    }
  }
},
}
```





```
"fiscalNumber": {
  "type": "string",
  "pattern": "TINIT-[A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}"
},
"idCard": {
  "type": "object",
  "properties": {
    "idCardType": {
      "type": "string"
    },
    "idCardDocNumber": {
      "type": "string"
    },
    "idCardIssuer": {
      "type": "string"
    },
    "idCardIssueDate": {
      "type": "string",
      "format": "date"
    },
    "idCardExpirationDate": {
      "type": "string",
      "format": "date"
    }
  }
},
"required": [
  "idCardType",
  "idCardDocNumber",
  "idCardIssuer",
  "idCardIssueDate",
  "idCardExpirationDate"
],
"mobilePhone": {
  "type": "object",
  "properties": {
    "countryCallingCode": {
      "type": "string",
      "pattern": "\\+[0-9]{2,4}"
    },
    "phoneNumber": {
      "type": "string",
      "pattern": "[0-9]{6,}"
    }
  }
},
"required": [
  "countryCallingCode",
  "phoneNumber"
]
```





```
,
"email": {
  "type": "string",
  "format": "email"
},
"address": {
  "type": "object",
  "properties": {
    "addressType": {
      "type": "string"
    },
    "addressName": {
      "type": "string"
    },
    "addressNumber": {
      "type": "string"
    },
    "postalCode": {
      "type": "string"
    },
    "municipality": {
      "type": "string"
    },
    "county": {
      "type": "string"
    },
    "nation": {
      "type": "string",
      "pattern": "Z[0-9]{3}"
    }
  }
},
"required": [
  "addressType",
  "addressName",
  "addressNumber",
  "postalCode",
  "municipality",
  "county",
  "nation"
]
},
"required": [
  "name",
  "familyName",
  "placeOfBirth",
  "countyOfBirth",
  "nationOfBirth",
  "dateOfBirth",
```





```
"gender",
"fiscalNumber",
"idCard",
"mobilePhone",
"email",
"address"
]
},
"optionalAttributes": {
"type": "object",
"properties": {
"digitalAddress": {
"type": "string"
}
}
}
},
"required": [
"mandatoryAttributes"
]
}
}
}
```

10.2 Json Schema per *token completo*

Il Json Schema utile per la validazione del *token completo* è illustrato nella tabella A.2

Tabella A.2

```
{
"definitions": {},
"$schema": "http://json-schema.org/draft-07/schema#",
"type": "object",
"required": [
"iss",
"sub",
"jti",
"aud",
"iat",
"exp",
"fiscalNumber",
"encryptedData"
],
"properties": {
"iss": {
"type": "string"
}
},
}
```





```
"sub": {
  "type": "string"
},
"jti": {
  "type": "string"
},
"aud": {
  "type": "string"
},
"iat": {
  "type": "string",
  "format": "date-time"
},
"exp": {
  "type": "string",
  "format": "date-time"
},
"fiscalNumber": {
  "type": "string",
  "pattern": "[A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}"
},
"encryptedData": {
  "type": "string"
}
}
```

10.3 Json Schema per Response Payload

Il Json Schema utile per la validazione payload della risposta dell'IdP all'invio di un *token completo* è illustrato nella tabella A.3

Tabella A.3

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "type": "object",
  "required": [
    "iss",
    "sub",
    "jti",
    "aud",
    "iat",
    "responseCode",
    "responseMessage"
  ],
  "properties": {
```





```
"iss": {
  "type": "string"
},
"sub": {
  "type": "string"
},
"jti": {
  "type": "string"
},
"aud": {
  "type": "string"
},
"iat": {
  "type": "string",
  "format": "date-time"
},
"responseCode": {
  "type": "number",
  "minimum": 1
},
"errorMessage": {
  "type": "string"
}
}
}
```

10.4 Riferimenti PKI AgID

I certificati di CA e SubCA e la CRL per la verifica della validità del Token sono disponibili ai seguenti indirizzi URL:

Certificato di CA Root

https://eidas.agid.gov.it/certificati/AgID_eIDAS_Root_CA.cer

Certificato di Subordinate CA per IdP

https://eidas.agid.gov.it/certificati/Sub_CA_SPID_IdP.cer

Certificato di Subordinate CA per Enti Pubblici

https://eidas.agid.gov.it/certificati/Sub_CA_SPID_pubblici.cer

Certificate Revocation List

https://eidas.agid.gov.it/crl/crl_SPID_Pub_SA.crl

