AGID | Agenzia per l'Italia Digitale

Linee Guida OpenID Connect in SPID

Versione 0.1 del 24/11/2021

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

Sommario

Capitolo 1

Introduzione

1.1 Scopo

Le Linee Guida vengono emesse ai sensi dell'articolo 71 del decreto legislativo 7 marzo 2005, n. 82 e successive modifiche e integrazioni (di seguito CAD) e della Determinazione AgID n. 160 del 2018 recante «Regolamento per l'adozione di linee guida per l'attuazione del Codice dell'Amministrazione Digitale».

OpenID Connect è un layer di identità basato su JSON/REST che si posiziona sopra al protocollo OAuth 2.0. La sua filosofia di design è "rendi semplici le cose semplici e rendi possibili le cose complicate". Mentre OAuth 2.0 è un protocollo di delega delle autorizzazioni di accesso generico, consentendo così il trasferimento di dati, e non definisce i modi per autenticare gli utenti o comunicare informazioni su di essi, OpenID Connect offre un layer di identità sicuro, flessibile e interoperabile in modo che le identità digitali possano essere facilmente utilizzate su servizi desktop e mobile.

OpenID Connect non si occupa solo di autenticazione ma può essere anche utilizzato per autorizzazione, delega e API access management.

I suoi punti di forza sono:

- facilità di integrazione;
- abilità di integrare applicazioni su diverse piattaforme, single-page app, web, backend, mobile, IoT;
- integrazione di componenti di terze parti in modalità sicura, interoperabile e scalabile;
- soluzione di diverse problematiche di sicurezza riscontrate in OAuth 2.0;
- utilizzo da parte di un gran numero di servizi social e di pagamento.

Per tutti questi motivi, le presenti linee guide intendono normare l'utilizzo di OpenID Connect nel Sistema Pubblico di Identità Digitale italiano (SPID).

1.2 Gruppo di lavoro

La redazione del documento è stata curata dal gruppo di lavoro composto da:

- Agenzia per l'Italia Digitale;
- Agenzia delle Entrate;
- Aruba S.p.A.;
- Comune di Roma;
- CSI Piemonte;
- Infocert S.p.A.;
- INPS;
- In.Te.S.A. S.p.A.;
- Istituto Poligrafico e Zecca dello Stato;
- Lepida S.p.A.;
- Lombardia Informatica S.p.A.;
- Namirial S.p.A.;
- Net Studio S.p.A.;
- Poste Italiane S.p.A.;
- Regione Toscana;
- Register.it S.p.A.;
- Sielte S.p.A.;
- Sistemi Informativi S.r.l.;
- Sogei S.p.A.;
- Team per la Trasformazione Digitale;
- TI Trust Technologies S.r.l.

I soggetti destinatari delle presenti linee guida sono: i Gestori dell'identità digitale e i Fornitori di servizi di cui al DPCM 24 ottobre 2014, "Definizione delle caratteristiche del sistema pubblico per la gestione dell'identità digitale di cittadini e imprese (SPID), nonché dei tempi e delle modalità di adozione del sistema SPID da parte delle pubbliche amministrazioni e delle imprese".

L'applicazione delle presenti linee guida è individuata come segue:

- per i gestori di identità digitali, l'obbligo di attuazione delle predette linee guida decorre dal 1 maggio 2022;

- per i fornitori di servizi, la facoltà di presentare domanda di adesione a SPID sulla base delle predette linee guida decorre dal 2 maggio 2022.

Capitolo 2

Riferimenti e sigle

2.1 Riferimenti Normativi

- [Reg. UE n. 910/2014] Regolamento (UE) n. 910/2014 del 23 luglio 2014 in materia di identificazione elettronica e servizi fiduciari per le transazioni elettroniche nel mercato interno e che abroga la direttiva 1999/93/CE;
- [D.Lgs. 82/2005] Decreto legislativo 7 marzo 2005, n. 82 e s.m.i. recante "Codice dell'amministrazione digitale";
- [D.P.C.M. 24 ottobre 2014] recante "Definizione delle caratteristiche del sistema pubblico per la gestione dell'identità digitale di cittadini e imprese (SPID), nonché dei tempi e delle modalità di adozione del sistema SPID da parte delle pubbliche amministrazioni e delle imprese.";
- [Regolamento recante le regole tecniche] (articolo 4, comma 2, DPCM 24 ottobre 2014) Determinazione AGID N. 44/2015 e s.m.i.;
- [Regolamento recante le modalità attuative per la realizzazione dello SPID] (articolo 4, comma 2, DPCM 24 ottobre 2014) Determinazione AGID N. 44/2015 e s.m.i.
- [GDPR] Regolamento (UE) 2016/679 e [Codice Privacy] Decreto legislativo 30 giugno 2003, n. 196 e s.m.i.

2.2 Standard di riferimento

SPID OpenID Connect è basato sul profilo iGov (openid-gov-profile) di OpenID Connect, *International Government Assurance Profile (iGov) for OpenID Connect 1.0*, con la seguente personalizzazione:

- paragrafo 4,2 di openid-igov-openid-connect-1_0: Scope: viene utilizzato solo lo scope "openid" e non "bio", "profile" e "doc" come suggerito dal profilo iGov;
- paragrafo 3,7 e 2,5 di openid-igov-openid-connect-1_0: I metadata degli attori sono distribuiti secondo le modalità definite dall'Agenzia per l'Italia Digitale.

Elenco dei riferimenti presenti nelle Linee Guida:

1. https://openid.net/specs/openid-igov-openid-connect-1_0-ID1.html
2. https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.1
3. https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.2
4. https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.4
5. https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.3.1
6. https://openid.net/specs/openid-igov-oauth2-1_0-03.html
7. https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.2.1.1
8. https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.2.1.2
9. https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.3.1.7
10. https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.3.2.2
11. https://openid.net/specs/openid-connect-core-1_0.html#AuthRequest
12. https://openid.net/specs/openid-connect-core-1_0.html#AuthRequestValidation
13. https://openid.net/specs/openid-connect-core-1_0.html#ClientAuthentication
14. https://openid.net/specs/openid-connect-core-1_0.html#FormSerialization
15. https://openid.net/specs/openid-connect-core-1_0.html#IDToken
16. https://openid.net/specs/openid-connect-core-1_0.html#IndividualClaimsRequests
17. https://openid.net/specs/openid-connect-core-1_0.html#JWTRequests
18. https://openid.net/specs/openid-connect-core-1_0.html#TokenEndpoint
19. https://openid.net/specs/openid-connect-core-1_0.html#TokenErrorResponse
20. https://openid.net/specs/openid-connect-core-1_0.html#UserInfoError
21. https://openid.net/specs/openid-connect-discovery-1_0.html#ProviderMetadata
22. https://openid.net/specs/openid-connect-registration-1_0.html#ClientMetadata
23. https://tools.ietf.org/html/rfc6749#section-3.2
24. https://tools.ietf.org/html/rfc6749#section-4.1.2
25. https://tools.ietf.org/html/rfc6749#section-4.1.2.1
26. https://tools.ietf.org/html/rfc6749#section-5.2
27. https://tools.ietf.org/html/rfc7009
28. https://tools.ietf.org/html/rfc7636
29. https://tools.ietf.org/html/rfc7662
30. https://tools.ietf.org/html/rfc7662#section-2.3
31. RFC8252: OAuth 2.0 for Native Apps (https://tools.ietf.org/html/rfc8252)

2.3 Termini e definizioni

Essendo le funzionalità simili, ritroviamo gli stessi concetti di SAML 2.0 anche in OpenID Connect:

SAML 2.0	OpenID Connect
Assertion	*ID Token*
Attribute query	*UserInfo Endpoint*
Authentication request	*Authentication request*
ForceAuthn	*prompt=login*
Identity Provider (IdP)	*OpenID Provider (OP)*
IdP metadata	*OpenID Provider metadata*
Issuer	*Issuer*
Logout	*Revoke*
NameID policy	*Subject identifier type*
Passive Authentication	*prompt=none*
Service Provider (SP)	*Relying Party (RP)*
SP metadata	*Client metadata*
Subject	*Subject Identifier*
Attributes	*Claims*

Ai fini delle presenti Linee Guida, per *OpenID Provider (OP)* e *Relying Party (RP)* si intendono rispettivamente i Gestori dell'identità digitale (Identity Provider - IdP) e i Fornitori di servizi (Service Provider - SP) di cui al DPCM 24 ottobre 2014, "Definizione delle caratteristiche del sistema pubblico per la gestione dell'identità digitale di cittadini e imprese (SPID), nonché dei tempi e delle modalità di adozione del sistema SPID da parte delle pubbliche amministrazioni e delle imprese".

Tutti gli esempi indicati nelle presenti LL.GG. non sono normativi.

Capitolo 3

Metadata

I metadata sono strutture dati contenenti le informazioni di OpenID Provider (OP) e di Relying Party (RP), mantenute e distribuite dal Registry SPID a tutti i soggetti della federazione, secondo le modalità definite dall'Agenzia per l'Italia Digitale, al fine di consentirne la configurazione nei rispettivi sistemi.

3.1 OpenID Provider (OP) Metadata

Il formato del metadata deriva da quanto specificato nel documento *"OpenID Connect Discovery 1.0"*, del quale costituisce un sottoinsieme con alcuni campi in aggiunta.

L'Agenzia per l'Italia Digitale definisce le modalità per l'uso alternativo di "jwks_uri" o di "jwks".

Esempio con jwks_uri:

```
{
 "issuer": "https://op.fornitore_identita.it",
 "authorization_endpoint": "https://op.fornitore_identita.it/auth",
 "token_endpoint": "https://op.fornitore_identita.it/token",
 "userinfo_endpoint": "https://op.fornitore_identita.it/userinfo",
 "introspection_endpoint": "https://op.fornitore_identita.it/introspect",
 "revocation_endpoint": "https://op.fornitore_identita.it/revoke",
 "end_session_endpoint": "https://op.fornitore_identita.it/logout",

 "jwks_uri": "https://registry.spid.gov.it/...",



 "id_token_encryption_alg_values_supported": [
     "..."
 ],
 "userinfo_signing_alg_values_supported": [
     "..."
 ],
 "request_object_encryption_enc_values_supported": [
     "..."
 ],
 "token_endpoint_auth_methods_supported": ["private_key_jwt"],
 "userinfo_encryption_alg_values_supported": [
     "..."
 ],
 "id_token_encryption_enc_values_supported": [
```

```
        "..."
    ],
    "id_token_signing_alg_values_supported": [
        "..."
    ],
    "request_object_encryption_alg_values_supported": [
        "..."
    ],
    "token_endpoint_auth_signing_alg_values_supported": [
        "..."
    ],
    "request_object_signing_alg_values_supported": [
        "..."
    ],
    "userinfo_encryption_enc_values_supported": [
        "..."
    ],
    "claims_supported":[
      "https://attributes.spid.gov.it/spidCode",
      "https://attributes.spid.gov.it/name",
      "https://attributes.spid.gov.it/familyName",
      "https://attributes.spid.gov.it/placeOfBirth",
      "https://attributes.spid.gov.it/countyOfBirth",
      "https://attributes.spid.gov.it/dateOfBirth",
      "https://attributes.spid.gov.it/gender",
      "https://attributes.spid.gov.it/companyName",
      "https://attributes.spid.gov.it/registeredOffice",
      "https://attributes.spid.gov.it/fiscalNumber",
      "https://attributes.spid.gov.it/ivaCode",
      "https://attributes.spid.gov.it/idCard",
      "https://attributes.spid.gov.it/mobilePhone",
      "https://attributes.spid.gov.it/email",
      "https://attributes.spid.gov.it/address",
      "https://attributes.spid.gov.it/expirationDate",
      "https://attributes.spid.gov.it/digitalAddress"
    ],
    "acr_values_supported":[
      "https://www.spid.gov.it/SpidL1",
      "https://www.spid.gov.it/SpidL2",
      "https://www.spid.gov.it/SpidL3
    ],
    "request_parameter_supported": true,
    "subject_types_supported":["pairwise"],
    "op_name": "Agenzia per l'Italia Digitale",
    "op_name#en": "Agency for Digital Italy",
    "op_uri": "https://www.agid.gov.it",
    "op_uri#en": "https://www.agid.gov.it/en"
  }
```

Esempio di risorsa jwks recuperabile alla url indicata in jwks_uri:

```
    {"keys": [
      {
        "kty": "EC",
        "kid": "sig-ec256-0",
        "use": "sig",
        "crv": "P-256",
        "x": "2jM2df3IjB9VYQ0yz373-6EEot_1TBuTRaRYafMi5K0",
        "y": "h6Zlz6XReK0L-iu4ZgxlozJEXgTGUFuuDl7o8b_8JnM"
      },
      {
        "kty": "EC",
```

```
          "kid": "enc-ec256-0",
          "use": "enc",
          "crv": "P-256",
          "x": "QI31cvWP4GwnWIi-Z0IYHauQ4nPCk8Vf1BHoPazGqEc",
          "y": "DBwf8t9-abpXGtTDlZ8njjxAb33kOMrOqiGsd9oRxr0"
        }
        ]
      }
```

Elemento	Descrizione
issuer	L'identificatore dell'OP (con schema HTTPS), corrispondente all'URL base. Deve essere identico al valore di iss negli ID Token prodotti dall'OP. L'issuer corrisponde al entityID che viene utilizzato in SAML e che rappresenta la chiave univoca con cui è identificato il fornitore di identità.
authorization_endpoint	URL dell'Authorization Endpoint, al quale il Client viene reindirizzato per iniziare il flusso di autenticazione.
token_endpoint	URL del Token Endpoint, che il RP deve chiamare per scambiare il codice ricevuto al termine dell'autenticazione con un access_token.
userinfo_endpoint	URL dello UserInfo Endpoint, che il RP può chiamare per ottenere i claim autorizzati dall'utente.
introspection_endpoint	URL dell'Introspection Endpoint (v. più avanti) che restituisce informazioni su un token.
revocation_endpoint	URL del Revocation Endpoint (v. più avanti) che revoca un *refresh token* o un *access token* già rilasciato al RP chiamante.
jwks	Json array composto dai seguenti parametri: • *kty:* famiglia dell'algoritmo crittografico utilizzato • *alg:* algoritmo utilizzato • *use:* utilizzo della chiave pubblica per firma (sig) o encryption (enc) • *kid:* identificatore univoco della chiave, valorizzato come RFC7638 • *n:* modulus (standard pem) • *e:* esponente (standard pem)
jwks_uri	Url del registry dove è localizzato il jwks che è un json array composto dai seguenti parametri: • *kty:* famiglia dell'algoritmo crittografico utilizzato

	• *alg:* algoritmo utilizzato • *use:* utilizzo della chiave pubblica per firma (sig) o encryption (enc) • *kid:* identificatore univoco della chiave, valorizzato come RFC7638 • *n:* modulus (standard pem) • *e:* esponente (standard pem)
op_name	Nome dell'OpenID Provider. Può essere specificato in più lingue apponendo al nome dell'elemento il suffisso "#" seguito dal codice RFC5646. Un nome di default senza indicazione della lingua è sempre presente.
op_uri	URL dell'OpenID Provider. Può essere specificato in più lingue apponendo al nome dell'elemento il suffisso "#" seguito dal codice RFC5646. Un valore di default senza indicazione della lingua è sempre presente.
request_object_signing_alg_values_supported	Array contenente gli algoritmi di firma supportati per il JWS dei Request Object. L'OP deve supportare RS256 e può supportare anche altri algoritmi definiti in rfc7518 (3.1): https://tools.ietf.org/html/rfc7518#section-3.1
request_object_encryption_alg_values_supported	Array contenente gli algoritmi di cifratura (***alg***) supportati per il JWS dei Request Object, come definito in rfc7518 (4.1): https://tools.ietf.org/html/rfc7518#section-4.1
request_object_encryption_enc_values_supported	Array contenente gli algoritmi di cifratura (***enc***) supportati per il JWS dei Request Object, come definito in rfc7518 (5.1): https://tools.ietf.org/html/rfc7518#section-5.1
id_token_signing_alg_values_supported	Array contenente gli algoritmi di firma supportati per il JWS dell'ID Token. L'OP deve supportare RS256 e può supportare anche altri algoritmi definiti in rfc7518 (3.1): https://tools.ietf.org/html/rfc7518#section-3.1
id_token_encryption_alg_values_supported	Array contenente gli algoritmi di cifratura (***alg***) supportati per il JWS dell'ID Token, come definito in rfc7518 (4.1): https://tools.ietf.org/html/rfc7518#section-4.1

id_token_encryption_enc_values_supported	Array contenente gli algoritmi di cifratura (***enc***) supportati per il JWS dell'ID Token, come definito in rfc7518 (5.1): https://tools.ietf.org/html/rfc7518#section-5.1
userinfo_signing_alg_values_supported	Array contenente gli algoritmi di firma supportati per il JWS dell'UserInfo Endpoint. L'OP deve supportare RS256 e può supportare anche altri algoritmi definiti in rfc7518 (3.1): https://tools.ietf.org/html/rfc7518#section-3.1
userinfo_encryption_alg_values_supported	Array contenente gli algoritmi di cifratura (***alg***) supportati per il JWE dell'UserInfo Endpoint, come definito in rfc7518 (4.1): https://tools.ietf.org/html/rfc7518#section-4.1
userinfo_encryption_enc_values_supported	Array contenente gli algoritmi di cifratura (***enc***) supportati per il JWE dell'UserInfo Endpoint, come definito in rfc7518 (5.1): https://tools.ietf.org/html/rfc7518#section-5.1
token_endpoint_auth_methods_supported	Array contenente i metodi di autenticazione supportati dal Token Endpoint. Deve essere presente solo il valore **private_key_jwt**
acr_values_supported	Array contenente i livelli SPID supportati dall'OP, rappresentati come URI. Può contenere uno o più valori tra i seguenti: • **https://www.spid.gov.it/SpidL1** • **https://www.spid.gov.it/SpidL2** • **https://www.spid.gov.it/SpidL3**
request_parameter_supported	Valore booleano che indica se il parametro ***request*** è supportato dall'OP. Deve essere obbligatoriamente **true**.
subject_types_supported	Array contenente i tipi di Subject Identifier supportati dall'OP. Deve contenere ***pairwise***.

Riferimenti

https://openid.net/specs/openid-connect-discovery-1_0.html#ProviderMetadata

3.2 Client Metadata (Relying Party Metadata)

Il formato del metadata deriva da quanto specificato nel documento *" OpenID Connect Dynamic Client Registration 1.0"*, del quale costituisce un sottoinsieme con alcuni campi in aggiunta.

L'Agenzia per l'Italia Digitale definisce le modalità per l'uso alternativo di "jwks_uri" o di "jwks".

Esempio con jwks_uri:

```
{
    "client_id": "https://rp.spid.agid.gov.it",
    "redirect_uris": [
        "https://rp.spid.agid.gov.it/callback1/",
        "https://rp.spid.agid.gov.it/callback2/"
    ],


    "jwks_uri": "https://registry.spid.gov.it/...",


    "response_types": ["code"],
    "grant_types": ["authorization_code", "refresh_token"],
    "client_name": "Agenzia per l'Italia Digitale",
    "client_name#en": "Agency for Digital Italy"
}
```

Elemento	Descrizione
client_id	URI che identifica univocamente il RP come da Registro SPID.
redirect_uris	Array di URI di redirezione utilizzati dal client (RP). Deve esserci un match esatto tra uno degli URI nell'array e quello utilizzato nell'Authentication request. Non è ammesso l'uso dello schema http (è obbligatorio HTTPS); tuttavia gli URI possono seguire eventuali schemi custom (ad es. *myapp://*) al fine di supportare applicazioni mobili. *Alla luce della normativa vigente in tema di protezione dei dati personali, è opportuno che l'URL non contenga informazioni utili ad individuare lo specifico servizio a cui l'utente intende accedere. Si raccomanda dunque di reindirizzare verso un sistema di access management che a sua volta rimanderà l'utente allo specifico servizio.*
jwks	Json array composto dai seguenti parametri: • *kty:* famiglia dell'algoritmo crittografico utilizzato • *alg:* algoritmo utilizzato • *use:* utilizzo della chiave pubblica per firma (sig) o encryption (enc) • *kid:* identificatore univoco della chiave, valorizzato come RFC7638 • *n:* modulus (standard pem) • *e:* esponente (standard pem)

jwks_uri	Url del registry dove è localizzato il jwks contenente la chiave pubblica in formato JSON Web Key (JWK) e quindi composto dai seguenti parametri: • *kty:* famiglia dell'algoritmo crittografico utilizzato • *alg:* algoritmo utilizzato • *use:* utilizzo della chiave pubblica per firma (sig) o encryption (enc) • *kid:* identificatore univoco della chiave, valorizzato come RFC7638 • *n:* modulus (standard pem) • *e:* esponente (standard pem).
client_name	Nome del RP da visualizzare nelle schermate di autenticazione e richiesta di consenso. Può essere specificato in più lingue apponendo al nome dell'elemento il suffisso "#" seguito dal codice RFC5646. Un nome di default senza indicazione della lingua è sempre presente.
response_types	Array dei valori di response_type previsti da OAuth 2.0 che il client userà nelle richieste di autenticazione. Deve contenere il solo valore **code**.
grant_types	Array dei valori di grant_type previsti da OAuth 2.0 che il client userà nelle richieste al Token Endpoint. Deve contenere i soli valori **authorization_code** e **refresh_token**.

Riferimenti

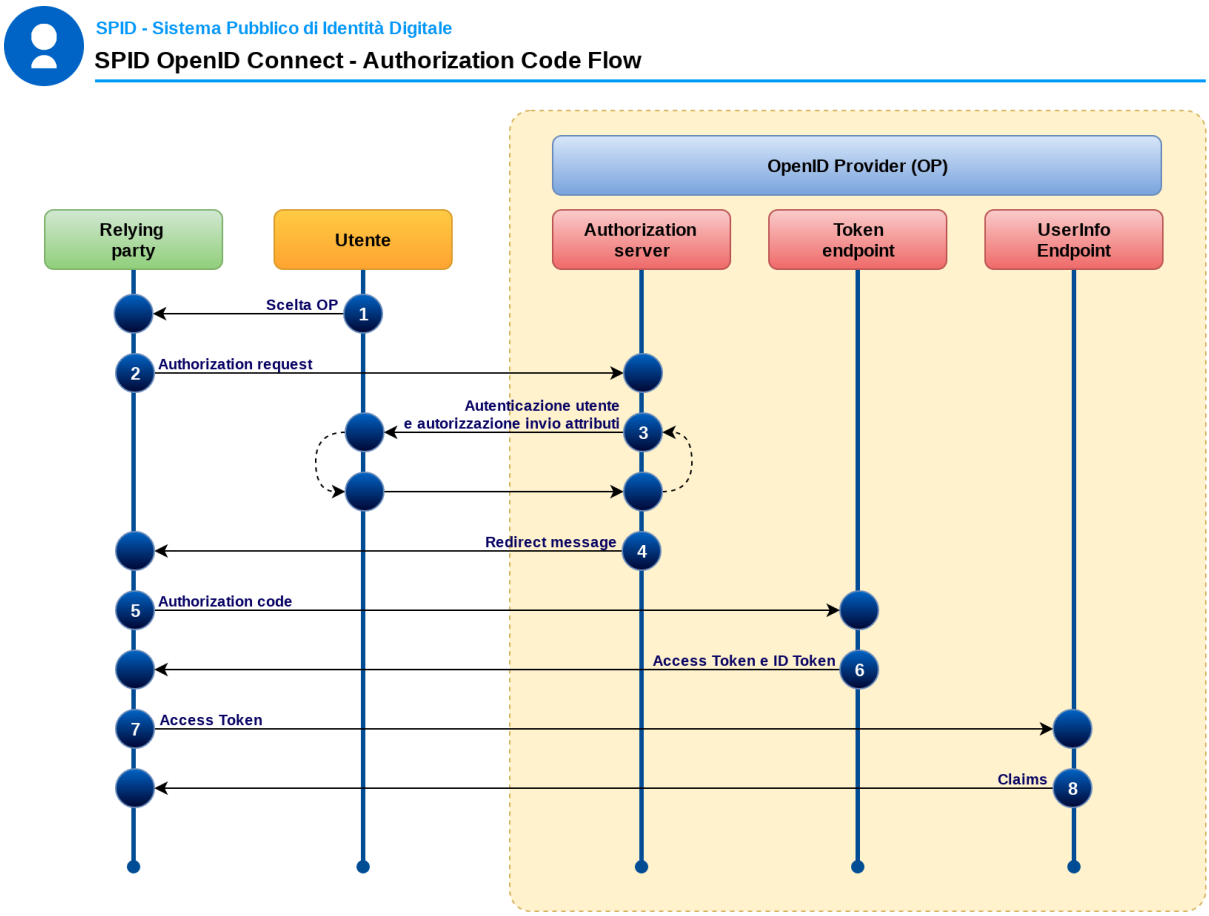
https://openid.net/specs/openid-connect-registration-1_0.html#ClientMetadata

Capitolo 4

Flusso

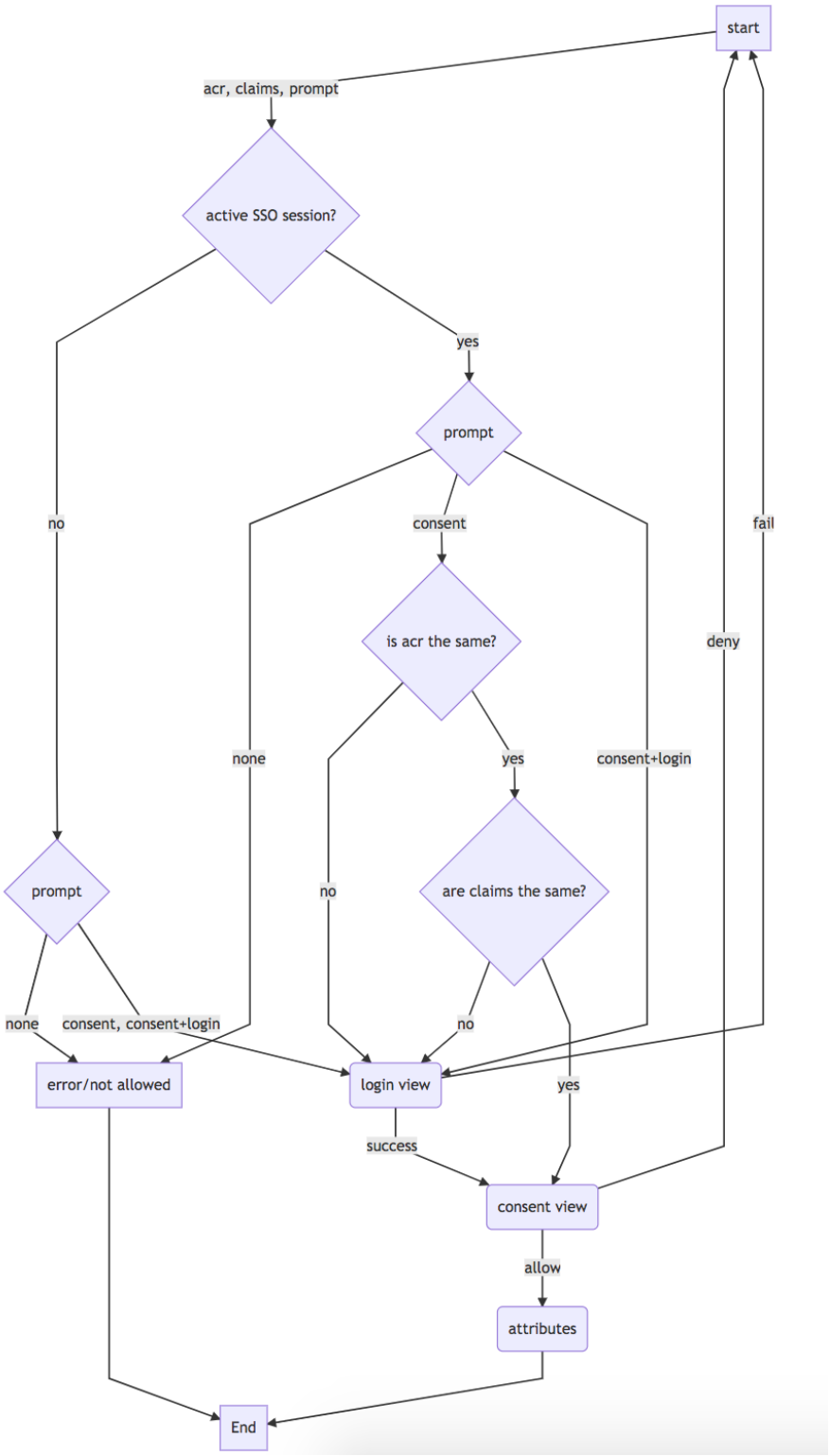
Il modello di flusso è l'"*OpenID Connect Authorization Code Flow*", che è l'unico flusso previsto da iGov.

L'Authorization code flow restituisce un codice di autorizzazione che può essere utilizzato per ottenere un ID token e/o un access token. Questo flusso è anche la soluzione ideale per sessioni lunghe o aggiornabili attraverso l'uso del refresh token. L'Authorization code flow ottiene l'authorization code dall'authorization endpoint dell'OpenID Provider e tutti i token sono restituiti dal token endpoint.



#	Da	A	Azione
1	**Utente**	**RP**	L'Utente, nella pagina di accesso del Relying Party (RP), seleziona, sul pulsante SPID, l'OpenID Provider (OP) con cui autenticarsi
2	**RP**	**OP Authorization server**	Il Relying Party (RP) prepara un'authentication request e reindirizza l'user agent dell'utente con l'authentication request verso l'Authorization Endpoint dell'OpendID Provider selezionato dall'utente
3	**OP Authorization server**	**Utente**	L'OpendID Provider (OP) richiede all'utente l'inserimento delle credenziali, secondo il livello SPID richiesto dal Relying Party (RP), all'utente a cui chiede, una volta autenticato, di autorizzare gli attributi richiesti dal Relying Party (RP)
4	**OP Authorization server**	**RP**	L'OpenID Provider reindirizza l'utente verso il Redirect URI specificato dal RP, passando un authorization code.
5	**RP**	**OP Token endpoint**	Il RP invia l'authorization code ricevuto al Token endpoint dell'OP
6	**OP Token endpoint**	**RP**	L'OP Token endpoint rilascia un ID Token, un Access token e se richiesto un Refresh token
7	**RP**	**UserInfo endpoint**	Il RP riceve e valida l'Access token e l'ID token. Per chiedere gli attributi che erano stati autorizzati dall'utente al punto 3, invia una richiesta all'UserInfo endpoint utilizzando l'Access token per l'autenticazione
8	**OP User endpoint**	**RP**	L'OP rilascia gli attributi richiesti

4.1 Conferma Utente invio dati al RP



4.2 Applicazioni per dispositivi mobili

Nel caso di applicazioni mobili rimane il requisito di seguire l'Authorization Code Flow descritto sopra.

In tale contesto, nel diagramma di cui al paragrafo precedente, l'elemento identificato come Relying Party sta ad indicare l'insieme dell'applicazione residente sul dispositivo mobile e del suo eventuale backend[1].

Le richieste al Token Endpoint e allo UserInfo Endpoint possono pertanto essere inviate sia dall'applicazione sia dal suo backend; lo scambio di informazioni tra l'applicazione mobile e il suo eventuale backend non sono normate dal presente documento, ferma restando la raccomandazione di prevedere meccanismi di trasmissione e archiviazione sicuri che impediscano a terze parti di venire in possesso dell'Access Token.

Per inviare la Authentication Request all'OP è possibile usare il browser o una webview, purché protetta con i meccanismi più sicuri messi a disposizione dai sistemi operativi al fine di ottenere il massimo isolamento dall'applicazione chiamante. A tal fine si consiglia l'uso della libreria AppAuth e si rinvia alle indicazioni contenute nelle Linee Guida User Experience SPID (Linee Guida UX SPID).

Si rimanda a RFC8252 per ulteriori specifiche tecniche e raccomandazioni di sicurezza da applicarsi in caso di applicazioni mobili.

Riferimenti

RFC8252: OAuth 2.0 for Native Apps (https://tools.ietf.org/html/rfc8252)

[1] Per Relying Party (RP) si intende sia il fornitore del servizio sia l'applicazione mediante la quale il fornitore erogare il Servizio, composta dal software installato sul device e dal server di "backend" gestito dal fornitore.

Capitolo 5

Authorization Endpoint (Authentication Request)

Per avviare il processo di autenticazione, il RP reindirizza l'utente all'Authorization Endpoint dell'OP selezionato, passando in POST o in GET una richiesta avente nel parametro **request** un oggetto in formato JWT.

Se viene utilizzato il metodo POST i parametri devono essere trasmessi utilizzando la Form Serialization (OIDC Connect Core 1.0 par. 13.2).

I parametri **client_id**, **response_type** e **scope** devono essere trasmessi sia come parametri sulla chiamata HTTP sia all'interno dell'oggetto request e i loro valori devono corrispondere, in ogni caso i parametri all'interno dell'oggetto request prevalgono su quelli indicati sulla chiamata HTTP.

L'oggetto request DEVE essere un token JWT firmato, secondo le modalità definite dall'Agenzia per l'Italia Digitale.

Esempio (chiamata HTTP):

```
https://op.spid.agid.gov.it/auth?client_id=https%3A%2F%2Frp.spid.agid.gov.it&
response_type=code&scope=openid& code_challenge=qWJlMe0xdbXrKxTm72EpH659bUxAxw80&
code_challenge_method=S256&request=eyJhbGciOiJSUzI1NiIsImtpZCI6ImsyYmRjIn0.ew0KIC
Jpc3MiOiAiczZCaGRSa3F0MyIsDQogImF1ZCI6ICJodHRwczovL3NlcnZlci5leGFtcGxlLmNvbSIsDQo
gInJlc3BvbnNlX3R5cGUiOiAiY29kZSBpZF90b2tlbiIsDQogImNsaWVudF9pZCI6ICJzNkJoZFJrcXQz
IiwNCiAicmVkaXJlY3RfdXJpIjogImh0dHBzOi8vY2xpZW50LmV4YW1wbGUub3JnL2NiIiwNCiAic2Nvc
GUiOiAib3BlbmlkIiwNCiAic3RhdGUiOiAiYWYwaWZqc2xka2oiLA0KICJub25jZSI6ICJuLTBTNl9Xek
EyTWoiLA0KICJtYXhfYWdlIjogODY0MDAsDQogImNsYWltcyI6IA0KICB7DQogICAidXNlcmluZm8iOiA
NCiAgICB7DQogICAgICJnaXZlbl9uYW1lIjogeyJlc3NlbnRpYWwiOiB0cnVlfSwNCiAgICAgI…
```

Esempio (contenuto del JWT):

```
{
    client_id=https%3A%2F%2Frp.spid.agid.gov.it
    response_type=code
    scope=openid
    code_challenge=qWJlMe0xdbXrKxTm72EpH659bUxAxw80
    code_challenge_method=S256
    nonce=MBzGqyf9QytD28eupyWhSqMj78WNqpc2
    prompt=login
    redirect_uri=https%3A%2F%2Frp.spid.agid.gov.it%2Fcallback1%2F
```

```
    acr_values=https://www.spid.gov.it/SpidL1 https://www.spid.gov.it/SpidL2
    claims={
             "id_token":{
                          "nbf": { essential: true},
                          "jti": { essential: true }
                        },
              "userinfo":{
                          "https://attributes.spid.gov.it/name": null,
                          "https://attributes.spid.gov.it/familyName": null
                         },
            }
    state=fyZiOL9Lf2CeKuNT2JzxiLRDink0uPcd
}
```

Parametro	Descrizione	Valori ammessi	Obbligatorio
client_id	URI che identifica univocamente il RP come da Registro SPID.	Deve corrispondere ad un valore nel Registro SPID.	SI
code_challenge	Un challenge per PKCE da riportare anche nella successiva richiesta al Token endpoint.	V. paragrafo 6.1 "Generazione del code_challenge per PKCE"	SI
code_challenge_method	Metodo di costruzione del challenge PKCE.	È obbligatorio specificare il valore **S256**	SI
nonce	Valore che serve ad evitare attacchi Reply, generato casualmente e non prevedibile da terzi. Questo valore sarà restituito nell'ID Token fornito dal Token Endpoint, in modo da consentire al client di verificare che sia uguale a quello inviato nella richiesta di autenticazione.	Stringa di almeno 32 caratteri alfanumerici.	SI

prompt	Definisce se l'OP deve occuparsi di eseguire una richiesta di autenticazione all'utente o meno.	**consent**: l'OP chiederà le credenziali di autenticazione all'utente (se non è già attiva una sessione di Single Sign-On) e successivamente chiederà il consenso al trasferimento degli attributi (valore consigliato). Se è già attiva una sessione di Single Sign-On, chiederà il consenso al trasferimento degli attributi. **consent login:** l'OP chiederà sempre le credenziali di autenticazione all'utente e successivamente chiederà il consenso al trasferimento degli attributi (valore da utilizzarsi limitatamente ai casi in cui si vuole forzare la riautenticazione). **verify**: l'OP verifica la presenza dell'utente tramite una prova di autenticazione, se è già attiva una sessione di Single Sign-On, e, successivamente, chiederà il consenso al trasferimento degli attributi. Se non è già attiva una sessione di Single Sign-On, l'OP chiederà le credenziali di autenticazione all'utente e, successivamente, chiederà il consenso al trasferimento degli attributi (valore facoltativo).	SI
redirect_uri	URL dove l'OP reindirizzerà l'utente al termine del processo di autenticazione.	Deve essere uno degli URL indicati nel client metadata (v. paragrafo 3.2).	SI
response_type	Il tipo di credenziali che deve restituire l'OP.	**code**	SI
scope	Lista degli scope richiesti.	**openid** (obbligatorio). **offline_access**: se specificato,	SI

		l'OP rilascerà oltre all'*access token* anche un *refresh token* necessario per instaurare sessioni lunghe revocabili. L'uso di questo valore è consentito solo se se si intende offrire all'utente una sessione lunga revocabile.	
acr_values	Valori di riferimento della classe di contesto dell'autenticazione richiesta. Stringa separata da uno spazio, che specifica i valori "acr" richiesti al server di autorizzazione per l'elaborazione della richiesta di autenticazione, con i valori visualizzati in ordine di preferenza. L'OP ha facoltà di utilizzare un'autenticazione ad un livello più alto di quanto richiesto. Tale scelta non deve comportare un esito negativo della richiesta.	https://www.spid.gov.it/SpidL1 https://www.spid.gov.it/SpidL2 https://www.spid.gov.it/SpidL3	SI
claims	Lista dei claims (attributi) che un RP intende richiedere.	v. paragrafo 5.1	SI
state	Valore univoco utilizzato per mantenere lo stato tra la request e il callback. Questo valore verrà restituito al	Stringa di almeno 32 caratteri alfanumerici.	SI

	client nella risposta al termine dell'autenticazione. Il valore deve essere significativo esclusivamente per il RP e non deve essere intellegibile ad altri.		
ui_locales	Lingue preferibili per visualizzare le pagine dell'OP. L'OP può ignorare questo parametro se non dispone di nessuna delle lingue indicate.	Lista di codici RFC5646 separati da spazi.	NO

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#FormSerialization
https://openid.net/specs/openid-connect-core-1_0.html#AuthRequest
https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.2.1.1
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.1
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.4
https://openid.net/specs/openid-connect-core-1_0.html#JWTRequests

5.1 Claims

Il parametro claims definisce gli attributi richiesti dal **RP**. Gli attributi SPID sono richiesti all'interno dell'elemento "*userinfo*", elencando gli attributi da richiedere come chiavi di oggetti JSON, i cui valori devono essere indicati come {*"essential": true*} o secondo le modalità definite

dall'Agenzia per l'Italia Digitale. Non è possibile richiedere attributi SPID nell'id_token. Gli attributi elencati sotto "userinfo" sono disponibili al momento della chiamata allo UserInfo Endpoint.

```
   {
"userinfo": {
    "https://attributes.spid.gov.it/familyName": {"essential": true}
     },
   }
```

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#IndividualClaimsRequests

5.2 Generazione del code challenge per PKCE

PKCE (Proof Key for Code Exchange, RFC7636) è un'estensione del protocollo OAuth 2.0 finalizzata ad evitare un potenziale attacco attuato con l'intercettazione dell'authorization code, soprattutto nel caso di applicazioni per dispositivi mobili. Consiste nella generazione di un codice (*code verifier*) e del suo hash (*code challenge*). Il *code challenge* viene inviato all'OP nella richiesta di autenticazione.

Quando il client contatta il Token Endpoint al termine del flusso di autenticazione, invia il *code verifier* originariamente creato, in modo che l'OP possa confrontare che il suo hash corrisponda con quello acquisito nella richiesta di autenticazione.

Il *code verifier* e il *code challenge* devono essere generati secondo le modalità definite dall'Agenzia per l'Italia Digitale

Riferimenti:

https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.3.1.7
https://tools.ietf.org/html/rfc7636

Capitolo 6

Authentication response

Un'Authentication response è un messaggio di risposta di autorizzazione OAuth 2.0 restituito dall'authorization endpoint dell'OpenID Provider (OP) al termine del flusso di autenticazione. L'OP reindirizzerà l'utente al redirect_uri specificato nella richiesta di autorizzazione, aggiungendo nella post i parametri in risposta.

Riferimenti:

https://tools.ietf.org/html/rfc6749#section-4.1.2
https://openid.net/specs/openid-connect-core-1_0.html#AuthRequestValidation

6.1 Response

Se l'autenticazione è avvenuta con successo, l'OpenID Provider (OP) Authorization server, reindirizza l'utente con i seguenti parametri:

```
https://rp.spid.agid.gov.it/resp?
  code=usDwMnEzJPpG5oaV8x3j&
  state=fyZiOL9Lf2CeKuNT2JzxiLRDink0uPcd
```

Parametro	Descrizione	Valori ammessi
code	Codice univoco di autorizzazione (*authorization code*) che il client poi passerà al Token Endpoint, secondo le modalità definite dall'Agenzia per l'Italia Digitale.	
state	Valore state incluso nell'Authentication request. Il client è tenuto a verificarne la corrispondenza.	Deve essere lo stesso valore indicato dal client nella Authorization Request.

6.2 Errori

In caso di errore, l'OP visualizza i messaggi di anomalia relativi agli scambi OpenID Connect descritti nelle relative tabelle definite dalle Linee Guida UX SPID. Nei casi in cui tali linee guida prescrivono un redirect dell'utente verso il RP, l'OP effettua il redirect verso l'URL indicata nel parametro ***redirect_uri*** della richiesta (solo se valido, ovvero presente nel client metadata), con i seguenti parametri.

Esempio:

```
https://rp.spid.agid.gov.it/resp?
  error=invalid_request&
  error_description=request%20malformata&
  state=fyZiOL9Lf2CeKuNT2JzxiLRDink0uPcd
```

Parametro	Descrizione	Valori ammessi
error	Codice dell'errore (v. tabella sotto)	
error_description	Descrizione più dettagliata dell'errore, finalizzata ad aiutare lo sviluppatore per eventuale debugging. Questo messaggio non è destinato ad essere visualizzato all'utente (a tal fine si faccia riferimento alle Linee Guida UX SPID).	
state	Valore *state* incluso nella Authentication Request.	Il client è tenuto a verificare che corrisponda a quello inviato nella Authentication Request.

Riferimenti:

https://tools.ietf.org/html/rfc6749#section-4.1.2.1

Capitolo 7

Token Endpoint (richiesta token)

Il Token Endpoint rilascia *access token*, *ID Token* e *refresh token*; vi sono due scenari distinti in cui il client chiama il Token Endpoint:

1. al termine del flusso di autenticazione descritto nel paragrafo precedente, il Client chiama il Token Endpoint inviando l'Authorization code ricevuto dall'OP (code=usDwMnEzJPpG5oaV8x3j) per ottenere un *ID Token* e un *access token* (necessario per poi chiedere gli attributi/claim allo UserInfo Endpoint) ed eventualmente un refresh token (se è stata avviata una sessione lunga revocabile);
2. in presenza di una sessione lunga revocabile, il Client chiama il Token Endpoint inviando il *refresh token* in suo possesso per ottenere un nuovo *access token*.

Riferimenti:

https://tools.ietf.org/html/rfc6749#section-3.2
https://openid.net/specs/openid-connect-core-1_0.html#TokenEndpoint
https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.2.1.2
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.2

7.1 Request

L'unico metodo di autenticazione all'endpoint token previsto è il private_key_jwt (OIDC Connect Core 1.0 par. 9)

Esempio di richiesta con authorization code (caso 1):

```
POST https://op.spid.agid.gov.it/token?
client_id=https%3A%2F%2Frp.spid.agid.gov.it&
client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiw
ibmFtZSI6IlNQSUQiLCJhZG1pbiI6dHJ1ZX0.LVyRDPVJm0S9q7oiXcYVIIqGWY0wWQlqxvFGYswL…&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-
bearer&
code=usDwMnEzJPpG5oaV8x3j&
code_verifier=9g8S40MozM3NSqjHnhi7OnsE38jklFv2&
grant_type=authorization_code
```

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#ClientAuthentication

Esempio di richiesta con refresh token (caso 2):

```
POST https://op.spid.agid.gov.it/token?
client_id=https%3A%2F%2Frp.spid.agid.gov.it&
client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiw
ibmFtZSI6IlNQSUQiLCJhZG1pbiI6dHJ1ZX0.LVyRDPVJm0S9q7oiXcYVIIqGWY0wWQlqxvFGYswL…&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-
bearer&
grant_type=refresh_token&
refresh_token=8xLOxBtZp8
```

Parametro	Descrizione	Valori ammessi	Obbligatorio
client_id	URI che identifica univocamente il RP come da Registro SPID.		SI
client_assertion	JWT firmato con la chiave privata del Relying Party contenente i seguenti parametri: **iss**: Identificatore del RP registrato presso gli OP e che contraddistingue univocamente l'entità nella federazione nel formato Uniform Resource Locator (URL); corrisponde al client_id usato nella richiesta di autenticazione **sub**: uguale al parametro **iss** **aud**: URL del Token Endpoint dell'OP		SI

	iat: data/ora in cui è stato rilasciato il JWT in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT). **exp**: data/ora di scadenza della request in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT). **jti**: Identificatore univoco per questa richiesta di autenticazione, generato dal client casualmente con almeno 128bit di entropia.	**iat**: secondo le modalità definite dall'Agenzia per l'Italia Digitale. **exp**: secondo le modalità definite dall'Agenzia per l'Italia Digitale.	
client_assertion_type		Deve assumere il seguente valore: **urn:ietf:params:oauth:client-assertion-type:jwt-bearer**	SI
code	Codice di autorizzazione restituito nell'Authentication response.		Solo se **grant_type** è **authorization_code**
code_verifier	Codice di verifica del code_challenge (v paragrafo 5.2)		Solo se **grant_type** è **authorization_code**
grant_type	Tipo di credenziale presentata dal Client per la richiesta corrente.	Può assumere uno dei seguenti valori: **authorization_code** **refresh_token**	SI
refresh_token			Solo se **grant_type** è **refresh_token**

7.2 Response

Dopo avere ricevuto e validato la Token request dal client, il Token endpoint dell'OpenID Provider (OP) restituisce una response che include ID Token e Access Token e un eventuale Refresh Token, in formato JWT e firmati secondo le modalità definite dall'Agenzia per l'Italia Digitale.

L'Access Token deve essere formatosecondo le indicazioni dello standard "International Government Assurance Profile (iGov) for OAuth 2.0 - Draft 03", paragrafo 3.2.1, "JWT Bearer Tokens".

L'ID Token deve essere formato secondo le indicazioni del paragrafo 7.3.

```
{
    "access_token": "dC34Pf6kdG...",
    "token_type": "Bearer",
    "refresh_token": "wJ848BcyLP...",
    "expires_in": 1800,
    "id_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY..."
}
```

Parametro	Descrizione	Valori ammessi
access_token	L'access token, in formato JWT firmato, consente l'accesso allo UserInfo endpoint per ottenere gli attributi.	
token_type	Tipo di *access token* restituito.	Deve essere valorizzato sempre con **Bearer**
refresh_token	Il *refresh token*, in formato JWT firmato, consente di chiamare nuovamente il Token Endpoint per ottenere un nuovo *access token* e quindi recuperare una sessione lunga revocabile.	
expires_in	Scadenza dell'*access token*, in secondi.	Secondo le modalità definite dall'Agenzia per l'Italia Digitale.
id_token	ID Token in formato JWT (v. paragrafo dedicato).	

7.3 ID Token

L'ID Token è un JSON Web Token (JWT) che contiene informazioni sull'utente che ha eseguito l'autenticazione. I Client devono eseguire la validazione dell'ID Token.

Esempio di *ID Token*:


```
{
    "iss": "https://op.spid.agid.gov.it/",
    "sub": "OP-1234567890",
    "aud": "https://rp.spid.agid.gov.it/auth",
    "acr": "https://www.spid.gov.it/SpidL2",
    "at_hash": "qiyh4XPJGsOZ2MEAyLkfWqeQ",
    "iat": 1519032969,
    "nbf": 1519032969,
    "exp": 1519033149,
    "jti": "nw4J0zMwRk4kRbQ53G7z",
    "nonce": "MBzGqyf9QytD28eupyWhSqMj78WNqpc2"
}
```


Parametro	Descrizione	Validazione
iss	Identificatore dell'OP che lo contraddistingue univocamente nella federazione nel formato Uniform Resource Locator (URL).	Il client è tenuto a verificare che questo valore corrisponda all'OP chiamato.
sub	Per il valore di questo parametro fare riferimento allo standard "OpenID Connect Core 1.0", paragrafo 8.1. "Pairwise Identifier Algorithm".	
aud	Contiene il client ID.	Il client è tenuto a verificare che questo valore corrisponda al proprio client ID.
acr	Livello di autenticazione effettivo. Può essere uguale o superiore a quello richiesto dal client nella Authentication Request.	

at_hash	Hash dell'Access Token; il suo valore è la codifica base64url della prima metà dell'hash del valore access_token, usando l'algoritmo di hashing indicato in ***alg*** nell'header dell'ID Token.	Il client è tenuto a verificare che questo valore corrisponda all'*access token* restituito insieme all'ID Token.
iat	Data/ora di emissione del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT).	
nbf	Data/ora di inizio validità del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT). Deve corrispondere con il valore di **iat**.	{ userinfo: {...} id_token: { acr: {...}, nbf: { essential: true}, jti: { essential: true } } }
exp	Data/ora di scadenza del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT), secondo le modalità definite dall'Agenzia per l'Italia Digitale.	
jti	Identificatore unico dell'ID Token che il client più utilizzare per prevenirne il riuso, rifiutando l'ID Token se già processato. Deve essere di difficile individuazione da parte di un attaccante e composto da una stringa casuale.	
nonce	Stringa casuale generata dal Client per ciascuna sessione utente ed inviata nell'Authentication Request (parametro nonce), finalizzata a mitigare attacchi replay.	Il client è tenuto a verificare che coincida con quella inviata nell'Authentication Request.

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#IDToken
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.3.1

7.4 Errori

In caso di errore, l'OP restituisce una **response** con un JSON nel body costituito dai parametri indicati nella tabella sottostante.

Esempio:

```
{
   "error": "codice errore",
   "error_description: "descrizione dell'errore"
}
```

Parametro	Descrizione	Valori ammessi
error	Codice dell'errore (v. tabella sotto)	
error_description	Descrizione più dettagliata dell'errore, finalizzata ad aiutare lo sviluppatore per eventuale debugging. Questo messaggio non è destinato ad essere visualizzato all'utente (a tal fine si faccia riferimento alle Linee Guida UX SPID).	

I codici di stato HTTP ed i valori dei parametri *error* e *error_description* sono descritti nelle tabelle relative ai messaggi di anomalia definiti dalle Linee Guida UX SPID.

Riferimenti:

https://tools.ietf.org/html/rfc6749#section-5.2
https://openid.net/specs/openid-connect-core-1_0.html#TokenErrorResponse

Capitolo 8

UserInfo Endpoint (attributi)

Lo UserInfo Endpoint è una risorsa protetta OAuth 2.0 che restituisce attributi dell'utente autenticato. Per ottenere gli attributi richiesti, il Relying Party inoltra una richiesta allo UserInfo endpoint utilizzando l'Access token.

Lo UserInfo Endpoint deve supportare l'uso del solo metodo HTTP GET [RFC2616], deve accettare il token di accesso, inviato all'interno del campo Authorization dell'Header, come token bearer OAuth 2.0 [RFC6750].

```
GET https://op.spid.agid.gov.it/userinfo
  Authorization: Bearer dC34Pf6kdG
```

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#UserInfo
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.4

8.1 Response

La response dello UserInfo Endpoint deve specificare nel "Content-Type" il valore "application/jwt".

Il contenuto trasmesso nel body della Response deve essere un JWT firmato e cifrato secondo le modalità definite dall'Agenzia per l'Italia Digitale.

Lo UserInfo Endpoint restituisce i claim autorizzati nella Authentication Request.

Esempio:


```
{
  "iss": "https://op.fornitore_identita.it",
  "aud": "https://rp.fornitore_servizio.it",
  "iat": 1519032969,
  "nbf": 1519032969,
  "exp": 1519033149,
  "sub": "OP-1234567890",
  "name": "Mario",
  "https://attributes.spid.gov.it/familyName": "Rossi",
  "https://attributes.spid.gov.it/fiscalNumber": "MROXXXXXXXXXXXXX"
}
```


Il payload del JWT è un JSON contenente i seguenti parametri:

Parametro	Descrizione	Valori ammessi
sub	Identificatore del soggetto, coincidente con quello già rilasciato nell'ID Token.	Il RP deve verificare che il valore coincida con quello contenuto nell'ID Token.
aud	Identificatore del soggetto destinatario della response (RP)	Il RP deve verificare che il valore coincida con il proprio client_id.
iss	URI che identifica univocamente l'OP.	
<attributo>	I claim richiesti al momento dell'autenticazione	

In caso di errore di autenticazione, lo UserInfo Endpoint restituisce un errore HTTP in accordo con quanto indicato nel par. 5.3.3., "UserInfo Error Response" di "OpenID Connect Core 1.0".

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#UserInfoError

Capitolo 9

Introspection Endpoint (verifica validità token)

L'Introspection Endpoint esposto dall'OP consente ai RP di ottenere informazioni su un token in loro possesso, come ad esempio la sua validità.

Riferimenti:

https://tools.ietf.org/html/rfc7662
https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.3.2.2

9.1 Request

La richiesta all'Introspection Endpoint consiste nell'invio del token su cui si vogliono ottenere informazioni unitamente a una Client Assertion che consente di identificare il RP che esegue la richiesta.

Esempio:

```
POST https://op.spid.agid.gov.it/introspection?
client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiw
ibmFtZSI6IlNQSUQiLCJhZG1pbiI6dHJ1ZX0.LVyRDPVJm0S9q7oiXcYVIIqGWY0wWQlqxvFGYswLF88…
&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-
bearer&
client_id=https%3A%2F%2Frp.spid.agid.gov.it&
token=eyJhbGciOiJSUzI1NiJ9.eyJleHAiOjE0MTg3MDI0MTQsImF1ZCI6WyJlNzFmYjcyYS05NzRmLT
QwMDEtYmNiNy1lNjdjMmJjMDAzN2YiXSwiaXNzIjoiaHR0cHM6XC9cL2FzLXZhLmV4YW1wbGUuY29tXC8
iLCJqdGkiOiIyMWIxNTk2ZC04NWQzLTQzN2MtYWQ4My1iM2YyY2UyNDcyNDQiLCJpYXQiOjE0MTg2OTg4
MTR9.FXDtEzDLbTHzFNroW7w27RLk5m0wprFfFH7h4bdFw5fR3pwiqejKmdfAbJvN3_yfAokBv06we5RA
RJUbdjmFFfRRW23cMbpGQCIk7Nq4L012X_1J4IewOQXXMLTyWQQ_BcBMjcW3MtPrY1AoOcfBOJPx1k2jw
RkYtyVTLWlff6S5gK-
ciYf3b0bAdjoQEHd_IvssIPH3xuBJkmtkrTlfWR0Q0pdpeyVePkMSI28XZvDaGnxA4j7QI5loZYeyzGR9
h70xQLVzqwwl1P0-F_0JaDFMJFO1yl4IexfpoZZsB3HhF2vFdL6D_lLeHRy-
H2g2OzF59eMIsM_Ccs4G47862w…
```

Parametro	Descrizione	Valori ammessi
client_assertion	JWT firmato con la chiave privata del Relying Party contenente gli stessi parametri documentati per le richieste al Token Endpoint.	L'OP deve verificare la validità di tutti i campi presenti nel JWT, nonché la validità della sua firma in relazione al parametro **client_id**.
client_assertion_type		**urn:ietf:params:oauth:client-assertion-type:jwt-bearer**
client_id	URI che identifica univocamente il RP come da Registro SPID.	L'OP deve verificare che il client_id sia noto.
token	Il token su cui il RP vuole ottenere informazioni.	

9.2 Response

L'Introspection Endpoint risponde con un oggetto JSON definito come segue.

Esempio:

```
{
  "active": true,
  "scope": "foo bar",
  "exp": 1519033149,
  "sub": "OP-1234567890",
  "client_id": https://rp.agid.gov.it/
  "iss": "https://op.spid.agid.gov.it/",
  "aud": "https://rp.spid.agid.gov.it/auth",
}
```

Parametro	Descrizione	Valori ammessi
active	Valore booleano che indica la validità del token. Se il token è scaduto, è revocato o non è mai stato emesso per il client_id chiamante, l'Introspection Endpoint deve restituire **false**.	

scope	Lista degli scope richiesti al momento dell'Authorization Request.	
exp	Scadenza del token.	
sub	Identificatore del soggetto, coincidente con quello già rilasciato nell'ID Token.	Il RP deve verificare che il valore coincida con quello contenuto nell'ID Token.
client_id	URI che identifica univocamente il RP come da Registro SPID.	Il RP deve verificare che il valore coincida con il proprio client_id.
iss	Identificatore dell'OP che lo contraddistingue univocamente nella federazione nel formato Uniform Resource Locator (URL).	Il client è tenuto a verificare che questo valore corrisponda all'OP chiamato.
aud	Contiene il client ID.	Il client è tenuto a verificare che questo valore corrisponda al proprio client ID.

9.3 Errori

In caso di errore, l'OP restituisce un codice HTTP 401 con un JSON nel body avente gli elementi di seguito indicati.

Esempio:

```
{
   "error": "invalid_client",
   "error_description: "client_id non riconosciuto."
}
```

Parametro	Descrizione	Valori ammessi
error	Codice dell'errore (v. tabella sotto)	
error_description	Descrizione più dettagliata dell'errore, finalizzata ad aiutare lo sviluppatore per eventuale debugging. Questo messaggio non è destinato ad essere visualizzato all'utente (a tal fine si faccia riferimento alle Linee Guida UX SPID).	

Di seguito i codici di errore:

Scenario	Codice errore
Il client_id indicato nella richiesta non è riconosciuto.	**invalid_client**
La richiesta non è valida a causa della mancanza o della non correttezza di uno o più parametri.	**invalid_request**
L'OP ha riscontrato un problema interno.	**server_error**
L'OP ha riscontrato un problema interno temporaneo.	**temporarily_unavailable**

Eventuali ulteriori codici di errore possono essere definiti dall'Agenzia per l'Italia Digitale con proprio atto.

Riferimenti:

https://tools.ietf.org/html/rfc7662#section-2.3

Capitolo 10

Revocation Endpoint (logout)

Il Revocation Endpoint consente al RP di chiedere la revoca di un *access token* o di un *refresh token* in suo possesso.

Quando l'utente esegue il logout o quando la sua sessione presso il RP scade (in base alle policy decise da quest'ultimo), il RP deve chiamare questo endpoint per revocare l'*access token* e l'eventuale *refresh token* in suo possesso.

L'OP dovrà revocare il token specificato nella richiesta e dovrà terminare la sessione di Single Sign-On se ancora attiva. Eventuali altri token attivi per l'utente dovranno invece essere mantenuti validi.

Riferimenti:

https://tools.ietf.org/html/rfc7009

10.1 Request

La richiesta al Revocation Endpoint consiste nell'invio del token che si vuole revocare unitamente a una Client Assertion che consente di identificare il RP che esegue la richiesta.

Esempio:

```
POST https://op.spid.agid.gov.it/revoke?
client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiw
ibmFtZSI6IlNQSUQiLCJhZG1pbiI6dHJ1ZX0.LVyRDPVJm0S9q7oiXcYVIIqGWY0wWQlqxvFGYswLF88&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-
bearer&
client_id=https%3A%2F%2Frp.spid.agid.gov.it&
token=eyJhbGciOiJSUzI1NiJ9.eyJleHAiOjE0MTg3MDI0MTQsImF1ZCI6WyJlNzFmYjcyYS05NzRmLT
QwMDEtYmNiNy1lNjdjMmJjMDAzN2YiXSwiaXNzIjoiaHR0cHM6XC9cL2FzLXZhLmV4YW1wbGUuY29tXC8
iLCJqdGkiOiIyMWIxNTk2ZC04NWQzLTQzN2MtYWQ4My1iM2YyY2UyNDcyNDQiLCJpYXQiOjE0MTg2OTg4
MTR9.FXDtEzDLbTHzFNroW7w27RLk5m0wprFfFH7h4bdFw5fR3pwiqejKmdfAbJvN3_yfAokBv06we5RA
RJUbdjmFFfRRW23cMbpGQCIk7Nq4L012X_1J4IewOQXXMLTyWQQ_BcBMjcW3MtPrY1AoOcfBOJPx1k2jw
RkYtyVTLWlff6S5gK-
ciYf3b0bAdjoQEHd_IvssIPH3xuBJkmtkrTlfWR0Q0pdpeyVePkMSI28XZvDaGnxA4j7QI5loZYeyzGR9
```

```
h70xQLVzqwwl1P0-F_0JaDFMJFO1yl4IexfpoZZsB3HhF2vFdL6D_lLeHRy-
H2g2OzF59eMIsM_Ccs4G47862w
```

Parametro	Descrizione	Valori ammessi
client_assertion	JWT firmato con la chiave privata del Relying Party contenente gli stessi parametri documentati per le richieste al Token Endpoint.	L'OP deve verificare la validità di tutti i campi presenti nel JWT, nonché la validità della sua firma in relazione al parametro **client_id**.
client_assertion_type		**urn:ietf:params:oauth:client-assertion-type:jwt-bearer**
client_id	URI che identifica univocamente il RP come da Registro SPID.	L'OP deve verificare che il client_id sia noto.
token	Il token su cui il RP vuole ottenere informazioni.	

10.2 Response

Il Revocation Endpoint risponde con un codice HTTP 200, anche nel caso in cui il token indicato non esista o sia già stato revocato (in modo da non rilasciare informazioni).

Capitolo 11

Sessioni lunghe revocabili

Per applicazioni mobili in cui l'RP intenda offrire un'esperienza utente che non passi per il reinserimento delle credenziali SPID ad ogni avvio, è possibile beneficiare di sessioni lunghe revocabili.

11.1 Ambiti e limiti di utilizzo

1. Al primo avvio dell'applicazione l'utente deve essere informato della possibilità di utilizzare la sessione lunga revocabile, per mantenere un'autenticazione di SPID di livello 1 che consenta all'applicazione di ricevere unicamente notifiche o call to action da parte dello SP, anche quando l'utente "non sia presente".
2. Le applicazioni mobili che fanno uso di sessioni lunghe revocabili sono tenute a richiedere all'utente, ad ogni avvio o attivazione, un PIN locale oppure un fattore biometrico memorizzato sul dispositivo dell'utente.
3. In fase di installazione o di prima configurazione, l'applicazione chiede all'utente di registrare il fattore di autenticazione da utilizzare per ogni avvio successivo al primo.
4. Quando l'utente avvia nuovamente l'applicazione, questa deve richiedere all'utente il fattore di autenticazione scelto in fase di installazione o di prima configurazione e consentire l'accesso alle funzioni del RP fruibili con il Livello 1 di SPID.
5. Nel caso in cui sia necessario accedere all'applicazione con un livello superiore a SPID di Livello 1, occorre effettuare una nuova autenticazione SPID in base al livello richiesto.

11.2 Request

Per poter utilizzare le sessioni lunghe revocabili, l'RP include nella Authentication Request:

- lo scope "offline_access", al fine di ottenere un refresh token utilizzabile dietro espressa consenso dell'utente.

Le sessioni lunghe sono consentite solo nel caso in cui nel parametro "acr_values" sia presente almeno il valore: https://www.spid.gov.it/SpidL1

11.3 Refresh Token

Se nella Request è incluso lo scope "offline_access" e il parametro "prompt" contiene tra i valori "consent", il Token Endpoint dell'OP restituisce oltre all'*access token* anche un *refresh token*.

11.4 Introspection

Ad ogni successivo avvio della propria applicazione, il RP può inviare una richiesta all'Introspection Endpoint per verificare che l'*access token* in suo possesso sia ancora valido.

In caso negativo, deve inviare una richiesta al Token Endpoint con il *refresh token* in suo possesso, per ottenere un nuovo *access token*.

Nel caso in cui il Token Endpoint rifiuti la concessione di un nuovo *access token*, l'utente dovrà effettuare un nuovo login SPID.

11.5 Esempio

Un RP fornisce servizi per i quali è necessaria un'autenticazione di liv 1 o di livello 2.

Il RP, per consentire l'accesso, effettua una richiesta di autenticazione, all'OP con acr_values=https://www.spid.gov.it/SpidL2 https://www.spid.gov.it/SpidL1

L'"authorization server" autentica l'utente, sulla base del Livello SPID richiesto dal RP (Livello 1 o Livello 2 o Livello 3), c.d. "autenticazione originaria", e rilascia un unico "access_token" sia per il Livello SPID 1 sia per il Livello SPID richiesto dal SP, con una scadenza di 15 minuti, e rilascia un "refresh_token" per il solo livello SPID 1 con scadenza 30 giorni.

L'OP consente l'accesso sia al livello "SPID1" sia al livello "SPID2" per 15 mins mediante l'"access_token".

Quando l'"access_token" scade, l'OP non consente l'accesso con tale l'access token e il RP deve ottenere un nuovo "access_token" tramite nuova autenticazione oppure tramite una "richiesta di refresh".

Il RP effettua una "richiesta di refresh" con il refresh_token.

Il "token endpoint" verifica la validità del refresh_token, e se nella richiesta di autenticazione originaria era presente nell'"acr_values" il livello "SPID1", rilascia un nuovo ID Token valido esclusivamente per il livello "SPID1" con scadenza a 30 giorni dall'autenticazione originaria.

Esempio (chiamata HTTP):

```
https://op.spid.agid.gov.it/auth?request=eyJhbGciOiJSUzI1NiIsImtpZCI6ImsyYmRjIn0.
ew0KICJpc3MiOiAiczZCaGRSa3F0MyIsDQogImF1ZCI6ICJodHRwczovL3NlcnZlci5leGFtcGxlLmNvb
SIsDQogInJlc3BvbnNlX3R5cGUiOiAiY29kZSBpZF90b2tlbiIsDQogImNsaWVudF9pZCI6ICJzNkJoZF
JrcXQzIiwNCiAicmVkaXJlY3RfdXJpIjogImh0dHBzOi8vY2xpZW50LmV4YW1wbGUub3JnL2NiIiwNCiA
ic2NvcGUiOiAib3BlbmlkIiwNCiAic3RhdGUiOiAiYWYwaWZqc2xka2oiLA0KICJub25jZSI6ICJuLTBT
Nl9XekEyTWoiLA0KICJtYXhfYWdlIjogODY0MDAsDQogImNsYWltcyI6IA0KICB7DQogICAidXNlcmluZ
m8iOiANCiAgICB7DQogICAgICJnaXZlbl9uYW1lIjogeyJlc3NlbnRpYWwiOiB0cnVlfSwNCiAgICAgI.
..
```

Esempio (contenuto del JWT):

```
{
    client_id=https://rp.spid.agid.gov.it,
    code_challenge=qWJlMe0xdbXrKxTm72EpH659bUxAxw80,
    code_challenge_method=S256,
    nonce=MBzGqyf9QytD28eupyWhSqMj78WNqpc2
    prompt=login,
    redirect_uri=https://rp.spid.agid.gov.it/callback1/,
    response_type=code,
    scope=openid offline_access,
    acr_values=https://www.spid.gov.it/SpidL1 https://www.spid.gov.it/SpidL2,
    claims={
        "userinfo":{
            "https://attributes.spid.gov.it/name": null,
            "https://attributes.spid.gov.it/familyName": null
        },
     },
    state=fyZiOL9Lf2CeKuNT2JzxiLRDink0uPcd
}
```

Parametro	Descrizione	Valori ammessi	Obbligatorio
client_id	URI che identifica univocamente il RP come da Registro SPID.	Deve corrispondere ad un valore nel Registro SPID.	SI
code_challenge	Un challenge per PKCE da riportare anche nella successiva richiesta al Token endpoint.	V. paragrafo 6.1 "Generazione del	SI

		code_challenge per PKCE"	
code_challenge_method	Metodo di costruzione del challenge PKCE.	È obbligatorio specificare il valore **S256**	SI
nonce	Valore che serve ad evitare attacchi Reply, generato casualmente e non prevedibile da terzi. Questo valore sarà restituito nell'ID Token fornito dal Token Endpoint, in modo da consentire al client di verificare che sia uguale a quello inviato nella richiesta di autenticazione.	Stringa di almeno 32 caratteri alfanumerici.	SI
prompt	Definisce se l'OP deve occuparsi di eseguire una richiesta di autenticazione all'utente o meno.	**consent**: l'OP chiederà le credenziali di autenticazione all'utente (ma solo se non è già attiva una sessione di Single Sign-On) e successivamente chiederà il consenso al trasferimento degli attributi (valore consigliato) **consent login:** l'OP chiederà sempre le credenziali di autenticazione all'utente e successivamente chiederà il consenso al trasferimento degli attributi (valore da utilizzarsi limitatamente ai casi in cui si vuole forzare la	SI

		riautenticazione)	
redirect_uri	URL dove l'OP reindirizzerà l'utente al termine del processo di autenticazione.	Deve essere uno degli URL indicati nel client metadata (v. paragrafo 3.2).	SI
response_type	Il tipo di credenziali che deve restituire l'OP.	**code**	SI
scope	Lista degli scope richiesti.	**openid** (obbligatorio) **offline_access**: se specificato, l'OP rilascerà oltre all'*access token* anche un *refresh token* necessario per instaurare sessioni lunghe revocabili. L'uso di questo valore è consentito solo se il client è un'applicazione per dispositivi mobili che intenda offrire all'utente una sessione lunga revocabile.	SI
claims	Lista dei claims (attributi) che un RP intende richiedere per il servizio.	v. paragrafo 5.1	SI
acr_values	Livello minimo SPID richiesto.	Se sono richiesti più livelli, occorre indicarli in ordine di preferenza separati da uno spazio.	SI
state	Valore univoco utilizzato per mantenere lo stato tra la request e il callback. Questo valore verrà restituito al client nella risposta al termine dell'autenticazione.	Stringa di almeno 32 caratteri alfanumerici.	SI

	Il valore deve essere significativo esclusivamente per il RP e non deve essere intellegibile ad altri.		
ui_locales	Lingue preferibili per visualizzare le pagine dell'OP. L'OP può ignorare questo parametro se non dispone di nessuna delle lingue indicate.	Lista di codici RFC5646 separati da spazi.	NO

Riferimenti:

https://openid.net/specs/openid-connect-core-1_0.html#AuthRequest
https://openid.net/specs/openid-igov-oauth2-1_0-03.html#rfc.section.2.1.1
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.1
https://openid.net/specs/openid-igov-openid-connect-1_0-03.html#rfc.section.2.4
https://openid.net/specs/openid-connect-core-1_0.html#JWTRequests

Esempio Refresh (chiamata HTTP):

```
POST /token HTTP/1.1
  Host: server.example.com
  Content-Type: application/x-www-form-urlencoded
    client_id=https%3A%2F%2Frp.spid.agid.gov.it
    &client_assertion=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3O
    DkwIiwibmFtZSI6IlNQSUQiLCJhZG1pbiI6dHJ1ZX0.LVyRDPVJm0S9q7oiXcYVIIqGWY0wWQlqx
    vFGYswLF88
    &client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-
    type%3Ajwt-bearer
    &grant_type=refresh_token
    &refresh_token=8xLOxBtZp8
```

Parametro	Descrizione	Valori ammessi
client_id	URI che identifica univocamente il RP come da Registro SPID.	Deve corrispondere alm valore del client_id della authentication request.
client_assertion	JWT firmato con la chiave privata del Relying Party contenente i seguenti parametri: **iss**: Identificatore del RP registrato presso gli OP e che	

	contraddistingue univocamente l'entità nella federazione nel formato Uniform Resource Locator (URL); corrisponde al client_id usato nella richiesta di autenticazione **sub**: uguale al parametro **iss** **aud**: URL del Token Endpoint dell'OP **iat**: data/ora in cui è stato rilasciato il JWT in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT) **exp**: data/ora di scadenza della request in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT). **jti**: Identificatore univoco per questa richiesta di autenticazione, generato dal client casualmente con almeno 128bit di entropia.	**iat**: secondo le modalità definite dall'Agenzia per l'Italia Digitale. **exp**: secondo le modalità definite dall'Agenzia per l'Italia Digitale.
client_assertion_type		**urn:ietf:params:oauth:client-assertion-type:jwt-bearer**
grant_type	Tipo di credenziale presentata dal Client per la richiesta corrente.	Deve assumere il valore: **refresh_token**
refresh_token		

Nel caso in cui il Token Endpoint rifiuti la concessione di un nuovo *ID token* e *access token*, l'utente dovrà effettuare un nuovo login SPID.

Nel caso in cui sia necessario accedere all'applicazione con un livello superiore a SPID di Livello 1, occorre effettuare una nuova autenticazione SPID in base al livello richiesto.

Se la Refresh Request è valida, l'OpenID Connect Provider restituisce un ID Token con i seguenti parametri:

Parametro	Descrizione	Valori ammessi
iss	Identificatore dell'OP che lo contraddistingue univocamente nella federazione nel formato Uniform Resource Locator (URL).	Deve essere lo stesso indicato nell'ID Token emesso nell'autenticazione originaria.
sub	Per il valore di questo parametro fare riferimento allo standard "OpenID Connect Core 1.0", paragrafo 8.1. "Pairwise Identifier Algorithm".	Deve essere lo stesso indicato nell'ID Token emesso nell'autenticazione originaria.
aud	Contiene il client ID.	Deve essere lo stesso indicato nell'ID Token emesso nell'autenticazione originaria.
acr	Livello di autenticazione ammesso a seguito di richiesta di refresh	https://www.spid.gov.it/SpidL1
at_hash	Hash dell'Access Token; il suo valore è la codifica base64url della prima metà dell'hash del valore access_token, usando l'algoritmo di hashing indicato in ***alg*** nell'header dell'ID Token.	Il client è tenuto a verificare che questo valore corrisponda all'*access token* restituito insieme all'ID Token.
iat	Data/ora di emissione del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT).	
nbf	Data/ora di inizio validità del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT). Deve corrispondere con il valore di **iat**.	
exp	Data/ora di scadenza del token in formato NumericDate, come indicato in RFC 7519 – JSON Web Token (JWT)	
jti	Identificatore unico dell'ID Token che il client può utilizzare per prevenirne il riuso, rifiutando l'ID Token se già processato. Deve essere di difficile individuazione da parte di un attaccante e composto da una stringa casuale.	

nonce	Stringa casuale generata dal Client per ciascuna sessione utente ed inviata nell'Authentication Request (parametro nonce), finalizzata a mitigare attacchi replay.	Il client è tenuto a verificare che coincida con quella inviata nell'Authentication Request.

Il refresh token ottenuto con la richiesta di autenticazione ha una validità massima di 30 giorni, entro i quali potrà essere utilizzato un numero illimitato di volte. Allo scadere dei 30 giorni non potrà più essere utilizzato e sarà necessario rieseguire l'autenticazione completa.

11.6 Gestione delle sessioni

Al fine di poter gestire le sessioni lunghe revocabili e poter rilasciare un refresh token per il Livello 1 di SPID anche a seguito di un'autenticazione di Livello 2 o 3 di SPID, è ammessa l'instaurazione, per ogni livello di SPID, di una sessione di autenticazione associata ad un determinato utente titolare di identità digitale, mantenuta dal gestore dell'identità digitale.

Gli OP devono includere all'interno della "Pagina di gestione dell'identità SPID", descritta nelle Linee Guida UX SPID, un'interfaccia per visualizzare le sessioni lunghe revocabili attive, dove l'utente possa revocarle singolarmente o in massa.

In caso di modifica della password richiesta dall'utente, l'OP deve prevedere la possibilità di revocare tutte le sessioni lunghe attive.

Capitolo 12

Gestione dei log

OpenID Provider e Relying party devono conservare i log di ogni autenticazion, che devono essere mantenuti per un tempo pari a 24 mesi.

In particolare devono essere conservate le evidenze di:

- rilascio di ID e access token a fronte di autenticazione;
- rilascio di refresh token a fronte di autenticazione;
- rilascio di ID e access token a fronte di utilizzo del refresh token.

Per ogni rilascio devono essere conservati JWT costituenti richiesta e risposta, occorre, inoltre, tracciare le chiamate e le relative risposte effettuate verso ogni endpoint.

Le tracciature dei log devono essere mantenute nel rispetto del GDPR e del Codice Privacy, sotto la responsabilità dell'OpenID Provider o del Relying Party, e l'accesso ai dati di tracciatura deve essere riservato a personale designato ai sensi dell'art. 2-quaterdecies del Codice Privacy.

Al fine di garantire la confidenzialità potrebbero essere adottati meccanismi di cifratura dei dati o impiegati sistemi di basi di dati (DBMS) che realizzano la persistenza cifrata delle informazioni.

Per il mantenimento devono essere messi in atto meccanismi che garantiscono l'integrità e il non ripudio dei log.