Sistema Federato Interregionale di Autenticazione: MODELLO ARCHITETTURALE DI RIFERIMENTO

Versione 1.0

INDICE

1. MODIFICHE AL DOCUMENTO

Descrizione modifica	Edizione	Data
TOC + traccia contenuti	0.1	15/02/2006
Estensione contenuti	0.2	05/03/2006
Revisione ed estensione contenuti	0.3	05/04/2006
Revisione ed estensione contenuti	0.4	07/04/2006
Revisione ed estensione contenuti	0.5	16/05/2006
Revisione ed estensione contenuti	0.6	31/05/2006
Revisione ed estensione contenuti	0.7	08/07/2006
Revisione ed estensione contenuti	0.9	17/07/2006
Revisione ed estensione contenuti	1.0	11/12/2006

2. INTRODUZIONE

Obiettivo del presente documento è la definizione e la descrizione del modello architetturale di riferimento del sistema federato interregionale di autenticazione. Il modello proposto prende spunto dal modello concettuale descritto in [2] e a partire da esso propone estensioni e raffinamenti in vista della realizzazione di un'implementazione che dimostri le funzionalità dell'infrastruttura di autenticazione federata di competenza dell'intervento infrastrutturale INF-3, come previsto dalla pianificazione del progetto ICAR [1].

La terminologia impiegata nel presente documento si conforma il più possibile a quella dei documenti di specifica del Sistema Pubblico di Cooperazione (SPCoop) definito dal CNIPA nell'ambito del SPC [4] nonché alla terminologia introdotta nel documento di modellazione concettuale [2].

Il presente documento è organizzato come segue. La sezione 3 riprende gli aspetti principali già descritti nel documento di modellazione concettuale e indica in che modo essi fungano da base per la modellazione architetturale. La sezione 4 illustra l'architettura proposta descrivendone i componenti, le relative interfacce, i modelli di interazione e le specifiche modalità d'impiego delle tecnologie coinvolte. La sezione 5 riassume le caratteristiche salienti del modello architetturale proposto e discute come esso si collochi globalmente nel contesto del progetto ICAR, soprattutto in termini di relazioni con gli altri interventi infrastrutturali, con i casi di studio applicativi e con le specifiche SPCoop. La sezione 6 discute le possibili evoluzioni del modello proposto.

3. DAL MODELLO CONCETTUALE AL MODELLO ARCHITETTURALE

Il modello concettuale di riferimento del sistema federato interregionale di autenticazione è definito in dettaglio in [2]. Nel documento citato si considera uno scenario tipico di una community network (per esempio una rete regionale) a cui afferiscono più domini in grado di offrire diversi servizi applicativi agli utenti finali. I servizi appartenenti a un dominio possono essere resi disponibili ad altri domini grazie a opportuni meccanismi di supporto alla federazione. Fattore abilitante per lo scenario descritto è la disponibilità di un sistema federato di autenticazione che supporti le politiche di sicurezza relative all'invocazione e alla fruizione dei servizi applicativi.

Volendo riassumere brevemente i principali contributi alla definizione del sistema federato di autenticazione che sono emersi durante la stesura del documento di modellazione concettuale, si può iniziare ricordando la definizione del concetto di dominio informatico e la sua declinazione nelle diverse tipologie di domini riconoscibili in uno scenario di interazione a livello di community network: un dominio rappresenta il sistema informativo di un'amministrazione, intesa in senso lato, e in particolare definisce il perimetro di sicurezza informatica di responsabilità di tale amministrazione. Il modello concettuale prevede quattro tipi distinti di domini (cfr. [2], sez. 3.1):

- di profilazione: è il dominio in cui è stata eseguita la procedura di riconoscimento iniziale di un determinato utente, durante la quale gli è stato richiesto di fornire alcune informazioni (attributi) che vengono poi memorizzate e archiviate in un'apposita struttura dati (il profilo utente);
- fruitore: è il dominio in cui ha origine la richiesta che dà luogo a un'interazione di accesso a un servizio offerto da un fornitore di servizi;
- erogatore: è il dominio in cui è presente il fornitore del servizio a cui l'utente intende accedere;
- certificatore: è il dominio in cui sono presenti uno o più soggetti (detti authority) in grado di certificare l'identità dell'utente così come il valore di uno o più attributi contenuti nel profilo utente. In particolare, il modello concettuale prevede i seguenti tipi di certificatori (cfr. [2], sez. 3.2):
  - Certification Authority: è l'entità abilitata a certificare l'identità di un utente;
  - Attribute Authority: è l'entità abilitata a certificare alcuni degli attributi contenuti nel profilo utente;
  - Profile Authority: è l'entità incaricata della gestione dei profili utente presenti nei domini di profilazione (in merito ai profili utente si veda [2], sez. 3.4).

Oltre ai certificatori, nei domini possono agire altre entità che è stato necessario modellare esplicitamente (cfr. [2], sez. 3.3): anzitutto gli utenti stessi che accedono ai servizi, che nella realtà sono però sempre mediati e quindi rappresentati da opportuni dispositivi chiamati User Agent (per esempio un comune browser web). Inoltre, per rendere possibile l'interazione di accesso ai servizi sono ovviamente necessari i fornitori di tali servizi, detti Service Provider. È da notare che, poiché anche i

Service Provider possono comportarsi da fruitori di servizi quando interagiscono con altri provider in cooperazione applicativa, essi possono essere globalmente considerati dei Service Requestor, categoria a cui appartengono di diritto anche gli utenti con i loro User Agent.

A supporto del sistema federato sono poi state introdotte e modellate altre entità (cfr. [2], sez. 3.3) quali l'Access Manager (o Local Proxy), che ha il compito di accedere alle varie authority o di mettere in contatto l'utente con esse per ottenere certificazioni di identità o di attributo, e il Gestore delle Politiche di Autorizzazione, incaricato di verificare che gli attributi certificati e ottenuti dal Local Proxy rispondano ai requisiti di accesso imposti dal fornitore del servizio (e opportunamente memorizzati in un profilo del servizio) per consentire o negare l'accesso all'utente.

Ciascuna delle entità elencate finora offre o richiede agli altri elementi del modello opportune interfacce che rendano possibile l'interazione e ne caratterizzino la natura e le funzionalità di base. Per questo motivo, il modello concettuale ha fornito una descrizione di alto livello di alcune delle interfacce offerte dalle principali entità identificate (per esempio cfr. [2], sez. 3.5).

Il passo successivo nella modellazione concettuale è stato rappresentato dalla puntuale e approfondita analisi dei modelli e degli scenari di interazione possibili tra le entità del modello (cfr. [2], sez. 4). Tale analisi aveva come obiettivo la selezione dello schema di interazione più opportuno rispetto ai requisiti del sistema federato di autenticazione (tenendo conto per esempio dei parametri relativi alla complessità del modello globale), cosa che ha guidato verso la minimizzazione del numero di entità in gioco in un'interazione completa, o la stima del minor carico richiesto alle diverse entità, specie se preesistenti, per l'adeguamento a quel determinato modello di interazione. In aggiunta, l'analisi ha evidenziato la necessità di introdurre nel modello nuove entità, come per esempio il Registry delle authority (cfr. [2], sez. 4.1). Questa attività ha infine condotto alla scelta del modello di riferimento (cfr. [2], sez. 4.5).

A valle di tale analisi è stato possibile procedere alla descrizione e alla definizione dei due principali scenari di riferimento per il sistema federato di autenticazione (cfr. [2], sez. 5), vale a dire:

- accesso di un utente a un servizio tramite web browser;
- cooperazione applicativa tra servizi nell'ambito dell'erogazione di un servizio a un utente.

La descrizione di questi due scenari è stata condotta in modo dettagliato, e ciò ha comportato la necessità di descrivere ad alto livello il modello dei dati a supporto della gestione federata delle identità degli utenti (per esempio il profilo utente e le asserzioni).

La sezione che conclude il documento di modellazione concettuale (cfr. [2], sez. 6) descrive le caratteristiche generali delle principali tecnologie correlate, cioè SAML (Security Assertion Markup Language) versione 2.0 e XACML (eXtensible Access Control Markup Language) versione 2.0, ne ipotizza l'utilizzo nel sistema federato di INF-3, e ne discute alcuni punti rilevanti ai fini dell'implementazione (per esempio il caching delle asserzioni).

Come si evince da questo pur sintetico richiamo, nel documento di modellazione concettuale si è considerato un contesto il più possibile esaustivo e rappresentativo di un sistema federato di autenticazione, includendo per completezza di trattazione anche elementi che non sono di stretta pertinenza dell'intervento infrastrutturale INF-3. Al fine di impostare correttamente la fase di

modellazione architetturale diventa importante stabilire con esattezza anzitutto quali aspetti tra quelli citati nel modello concettuale rientrino effettivamente tra le responsabilità di INF-3 (cioè quelli che nel presente documento verranno dettagliati quanto ad architettura interna, interfacce offerte e attese, protocolli di comunicazione, tecnologie correlate, linee guida per la realizzazione, ecc.) e quali invece ne risultino in ultima analisi esclusi (cioè quelli per i quali ci si limiterà in questa sede a considerazioni di alto livello sulle interfacce e sulle tecnologie coinvolte).

Alla luce di queste considerazioni, il presente documento di modellazione architetturale si occuperà in particolare di:

- consolidare l'architettura di alto livello del sistema federato di autenticazione;
- definire l'architettura delle entità cardine del sistema federato di autenticazione, cioè il Local Proxy, la Profile Authority e l'Authority Registry;
- definire il modello dei dati adottato dalle entità descritte al punto precedente e i protocolli di comunicazione adoperati nelle interazioni con gli altri componenti del sistema, in particolare:
    - le tipologie di asserzioni scambiate (che includono statement di identità e di attributo) e i relativi protocolli di richiesta e risposta;
    - le modalità di interazione tra il Service Provider e il Local Proxy;
    - la struttura del profilo utente gestito dalla Profile Authority;
    - il protocollo di interrogazione della Profile Authority;
    - il protocollo di interrogazione dell'Authority Registry.
- descrivere in dettaglio le interazioni nei seguenti scenari di riferimento, già identificati nel documento di modellazione concettuale:
    - accesso dell'utente via web browser a un servizio offerto da un Service Provider;
    - fruizione in cooperazione applicativa da parte di un Service Provider di un servizio offerto da un altro Service Provider appartenente a un dominio differente, a seguito di una richiesta utente.

Altri aspetti, pur legati al funzionamento del sistema federato, non saranno invece oggetto del presente documento poiché non rientrano nelle responsabilità dell'intervento infrastrutturale INF-3. Tra questi aspetti figurano, per esempio:

- la definizione dell'architettura interna di un Service Provider, sulla quale ciascun fornitore di servizi deve poter mantenere completa autonomia; tuttavia verranno fornite indicazioni di massima sui principali sottocomponenti necessari da un punto di vista logico e sulle interfacce che dovranno essere esposte, con riferimento agli scenari già introdotti nel documento di modellazione concettuale;
- la definizione dell'architettura interna dello User Agent, sul quale è sufficiente fare l'ipotesi che si tratti di un client web (per esempio un comune web browser);

- la definizione dell'architettura interna di una Certification Authority o di una Attribute Authority, poiché tali entità sono del tutto autonome rispetto al sistema federato di autenticazione (si fornirà però una descrizione delle interfacce che dovranno essere esposte);
- la definizione di meccanismi di gestione delle "identità" dei servizi applicativi (identificazione univoca, autenticazione, ecc.), in quanto l'intervento infrastrutturale INF-3 riguarda l'autenticazione basata esclusivamente sull'identità di utenti umani;
- la descrizione dettagliata di altri possibili scenari di interazione tra utenti e servizi, per esempio la gestione dell'autenticazione in caso di accesso di un utente a un servizio mediante interazione applicativa (per esempio l'invocazione di un servizio esposto come web service attraverso un'applicazione client dell'utente).

Nel prosieguo del presente documento saranno pertanto approfonditi gli aspetti architetturali di dettaglio relativi alle entità che rientrano nei confini di competenza dell'intervento infrastrutturale INF-3.

Si fa notare che, per omogeneità di terminologia rispetto allo standard SAML, nel presente documento si è ritenuto opportuno rinominare l'entità Certification Authority del modello concettuale in Identity Provider.

4. ARCHITETTURA

L'obiettivo di questa sezione è fornire una descrizione architetturale dettagliata del sistema federato di autenticazione e illustrare come vengono utilizzate al suo interno le tecnologie di riferimento (in particolare SAML, cfr. sez. 4.2). Si noti che alcuni aspetti non verranno illustrati in dettaglio ma solo richiamati brevemente, in quanto già trattati esaurientemente nel modello concettuale di riferimento. Per essi si rimanda pertanto al relativo documento [2] e a quanto riportato nella sez. 3 del presente documento.

4.1. Vista d'insieme

Il diagramma di Figura 1 illustra ad alto livello l'architettura del sistema. In particolare sono mostrate le entità coinvolte nell'erogazione dei servizi di autenticazione federata INF-3 e le relative interfacce (non sono qui mostrate altre interfacce non direttamente pertinenti al funzionamento del sistema federato). N.B.: i rettangoli che circondano le entità rappresentate nel diagramma circoscrivono intuitivamente gli "ambiti di competenza" dei componenti rispetto alle responsabilità di modellazione e realizzazione assegnate all'interno del progetto ICAR. In particolare, il riquadro "INF-3" racchiude le entità che saranno maggiormente dettagliate in questa sezione, sia in termini di struttura interna che di interfacciamento con il resto del sistema. Si sottolinea che questa suddivisione non vuole fornire indicazioni stringenti in merito al dispiegamento dei componenti rappresentati.

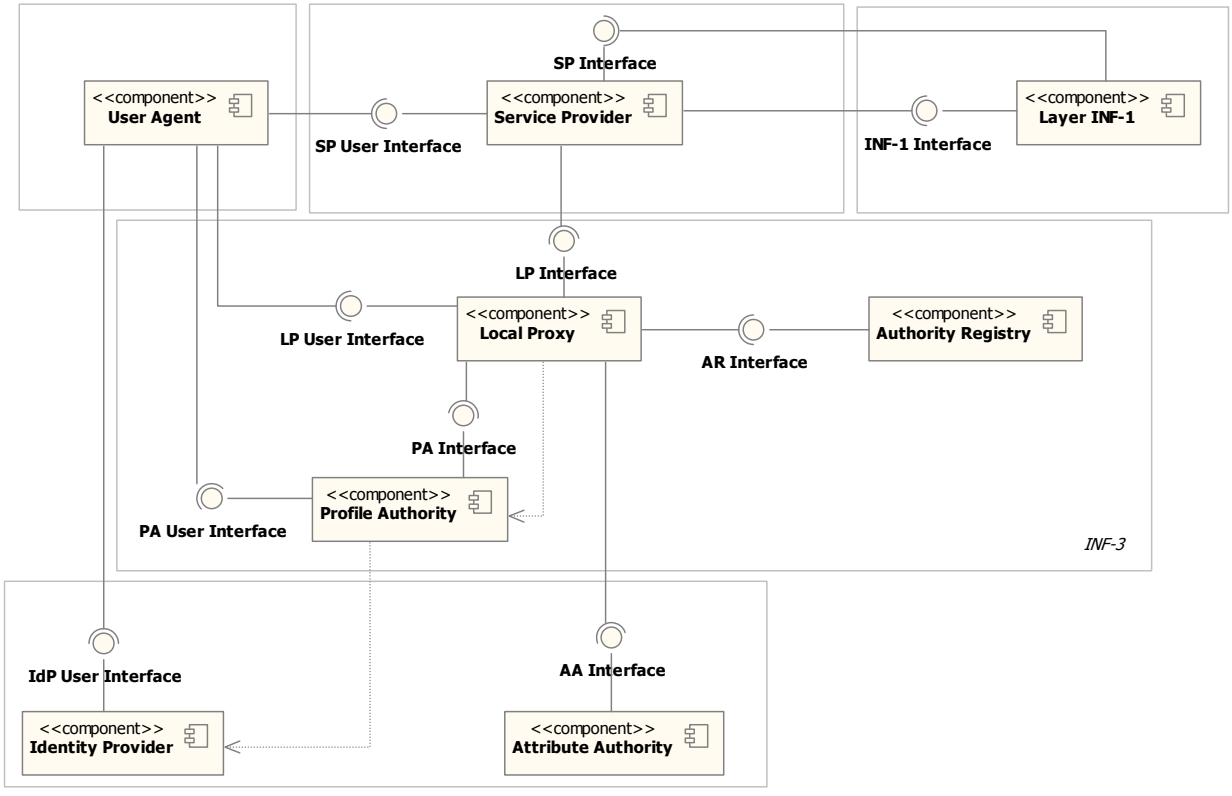

Figura 1. Vista architetturale d'insieme del sistema federato di autenticazione

Si fornisce qui di seguito una sintetica descrizione di ciascun componente e delle interfacce offerte. Maggiori dettagli relativi a ciascun componente (architettura interna, interfacce attese, protocolli di interazione e modello dei dati, ecc.) saranno trattati in sezioni dedicate (dalla sez. 4.3.1 alla sez. 4.3.8). Inoltre, per ulteriori chiarimenti sul ruolo e sul comportamento di ciascun componente si veda la descrizione degli scenari di riferimento (sez. 4.4).

Lo User Agent è un client web (per esempio un web browser) usato dall'utente per accedere ai servizi offerti dai Service Provider. Allo User Agent si richiede di supportare i protocolli HTTP e HTTPS con scambio mutuo di certificati tra client e server.

Per maggiori informazioni sullo User Agent si veda la sez. 4.3.1.

Il Service Provider è il fornitore dei servizi applicativi. Il Service Provider espone due interfacce:

- `SP User Interface`: permette agli utenti l'accesso via web tramite User Agent ai servizi offerti;
- `SP Interface`: permette l'interazione con altri Service Provider in modalità di cooperazione applicativa.

Per maggiori informazioni sul Service Provider si veda la sez. 4.3.2.

Il Local Proxy è il componente che, dal punto di vista del Service Provider, si comporta da proxy verso l'infrastruttura di autenticazione federata. Il componente espone due interfacce:

- `LP User Interface`: l'interfaccia utente per l'interazione con l'utente tramite User Agent (utilizzata per esempio nel caso del servizio "WAYF", con cui il Local Proxy chiede all'utente di indicare la sua Profile Authority di riferimento, cfr. sez. 4.4.1);
- `LP Interface`: l'interfaccia applicativa per l'interazione con altri componenti architetturali, per esempio il Service Provider.

Si notino le dipendenze d'uso esistenti tra il Local Proxy e la Profile Authority, e tra quest'ultima e l'Identity Provider, che verranno chiarite nel seguito. Per maggiori informazioni sul Local Proxy si veda la sez. 4.3.3.

L'Authority Registry è il componente che permette di recuperare gli URI e altre informazioni relative alle authority in base al loro nome logico. Esso espone un'interfaccia:

- `AR Interface`: l'interfaccia applicativa che supporta le operazioni di interrogazione del registro.

Per maggiori informazioni sull'Authority Registry si veda la sez. 4.3.4.

L'Identity Provider è il componente responsabile della certificazione dell'identità degli utenti. Esso espone un'interfaccia:

- `IdP User Interface`: l'interfaccia web destinata all'immissione delle credenziali di autenticazione da parte dell'utente.

Si osserva che nel modello concettuale questo componente era chiamato Certification Authority. Per maggiori informazioni sull'Identity Provider si veda la sez. 4.3.5.

La Profile Authority è il componente che svolge il ruolo di repository dei profili utente e che si occupa di interagire con l'Identity Provider quando l'utente si deve autenticare. Esso espone due interfacce:

- `PA Interface`: l'interfaccia applicativa che supporta le operazioni di interrogazione dei profili utente.
- `PA User Interface`: l'interfaccia web per l'interazione con l'utente (creazione e gestione dei profili utente, scelta dell'Identity Provider, selezione del profilo, ecc.).

In casi particolari questo componente può agire da Attribute Authority. Per maggiori informazioni sulla Profile Authority si veda la sez. 4.3.6.

La Attribute Authority è il componente in grado di certificare gli attributi presenti in un profilo utente. Esso espone un'interfaccia:

- `AA Interface`: l'interfaccia applicativa che supporta le operazioni di interrogazione per la validazione degli attributi utente.

Per ulteriori dettagli sulla Attribute Authority si veda la sez. 4.3.7.

Infine, il layer INF-1 permette l'interazione applicativa inter-dominio offrendo ai componenti di INF-3 (in particolare ai Service Provider) un'opportuna interfaccia:

- `INF-1 Interface`: l'interfaccia applicativa offerta ai Service Provider per la fruizione di altri servizi in cooperazione applicativa.

Alcune considerazioni su questo componente si trovano nella sez. 4.3.8.

4.2. Uso delle tecnologie di riferimento

Prima di passare alla descrizione di dettaglio di ciascun componente architetturale, in questa sezione si discutono le modalità con le quali le tecnologie di riferimento (soprattutto lo standard SAML [8]) saranno impiegate a supporto del modello dei dati del sistema federato di autenticazione.

4.2.1. Interfacce dei componenti architetturali

Nel modello architetturale descritto in questo documento si è operata la scelta di considerare ciascuno dei principali componenti architetturali come una SAML authority. Ciò conferisce uniformità alla trattazione delle interfacce offerte dai componenti, e comporta l'ulteriore vantaggio di fare riferimento a uno standard consolidato e normato quanto a struttura dei messaggi e protocolli di comunicazione. Inoltre, in questo modo si possono avere in risposta asserzioni firmate digitalmente dalle authority emittenti, poiché questa possibilità è già prevista dallo standard SAML.

In sintesi, questo equivale a dire che ciascun componente che espone un'interfaccia applicativa dev'essere in grado di accettare in ingresso un messaggio SAML di richiesta (per esempio una `<AuthnRequest>` o una `<AttributeQuery>`) e di generare in risposta un messaggio SAML (per esempio una `<Response>`) opportunamente strutturati per veicolare le informazioni di volta in volta necessarie ad attuare l'interazione (per esempio un'interrogazione diretta a una Attribute Authority, un Authority Registry o una Profile Authority, e la trasmissione delle relative risposte). Si

rende dunque necessario definire caso per caso il modo in cui è possibile interagire con ciascuna entità sfruttando la struttura dei messaggi SAML standard – o di loro opportune estensioni, specie nei casi in cui le entità in questione non coincidono direttamente con i ruoli SAML standard (è questo per esempio il caso dell'Authority Registry e, in certi casi, della Profile Authority).

In generale, lo standard SAML prevede l'uso del costrutto `<AuthnRequest>` per inoltrare richieste di autenticazione, e del costrutto `<AttributeQuery>` per inoltrare richieste di attributo. In quest'ultimo caso, il protocollo di richiesta permette di ottenere informazioni sugli attributi – e sui relativi valori – che si riferiscono a un determinato subject. Gli attributi di interesse possono essere tutti quelli relativi al subject specificato oppure un loro sottoinsieme. La selezione è possibile usando opportunamente il nome dell'attributo ed eventualmente anche il relativo elemento `<AttributeValue>`. In risposta si otterrà una `<Response>` contenente gli `<AttributeStatement>` relativi agli attributi richiesti. Per maggiori informazioni sulla struttura degli elementi citati si veda il documento [9], sez. 3.3.2.3 e 2.7.3.1.

Alla luce della specifica SAML, dunque, per mezzo dell'elemento `<AttributeQuery>` è possibile realizzare un'interfaccia base di registry lookup così come un'interfaccia di interrogazione di una Profile Authority senza ricorrere a particolari estensioni custom dello schema SAML. Questa è esattamente la strada seguita nel modello architetturale descritto nel presente documento. Nel nostro caso, però, per ragioni di generalità e di estendibilità si è scelto di non fare riferimento nella modellazione direttamente ai costrutti base previsti dal protocollo SAML, bensì di definire concettualmente delle opportune specializzazioni di tali costrutti che lascino spazio a evoluzioni e specializzazioni future del modello proposto, pur rimanendo pienamente compatibili con gli standard adottati (quindi senza precludere l'uso di implementazioni esistenti SAML-compliant). Attualmente la specializzazione introdotta non rappresenta quindi una vera e propria estensione del linguaggio, se non in pochi casi, quanto piuttosto una relazione "is-a" con in più un'indicazione sugli eventuali requisiti che i costrutti SAML adottati devono avere nel contesto di INF-3 (per esempio la presenza obbligatoria di alcuni elementi che la specifica considera opzionali).

La figura che segue mostra alcuni degli elementi SAML considerati nel contesto del modello architetturale di INF-3. La struttura e le modalità d'uso di tali elementi saranno precisate caso per caso nel seguito di questa sezione.

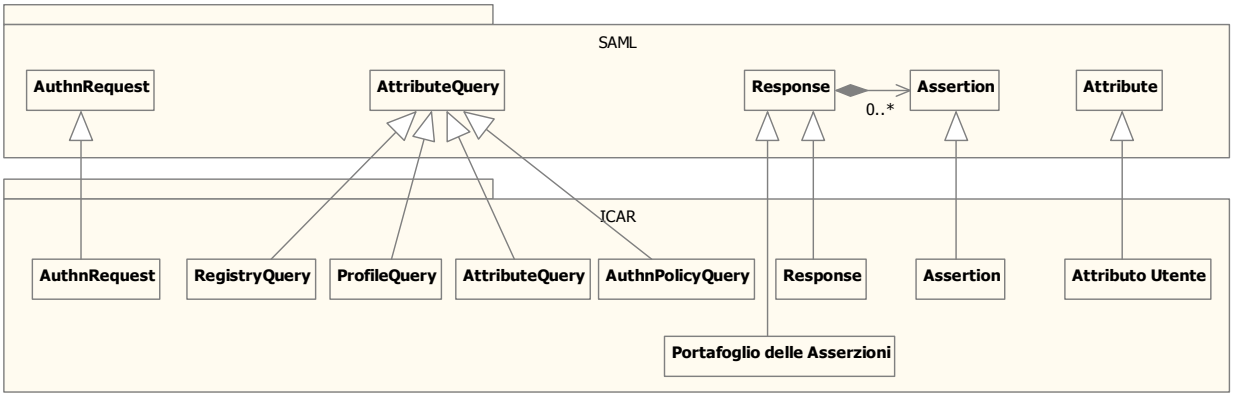

Figura 2. Uso di alcuni costrutti SAML in INF-3

4.2.2. *Meccanismi di autenticazione*

Gli scenari di interazione con l'utente a fini di autenticazione (che saranno descritti in dettaglio nella sez. 4.4) si mappano su alcuni profili standard SAML (cfr. [8], sez. 4). In particolare si considerano qui profili "Service Provider initiated", in cui cioè il meccanismo di autenticazione è innescato dalla richiesta inoltrata dall'utente al Service Provider, il quale a sua volta si rivolge opportunamente all'autorità di certificazione d'identità in modalità "pull". Attualmente non si considerano in dettaglio profili "Identity Provider initiated" in quanto meno significativi ai fini del modello federato di autenticazione rispetto ai precedenti: gli scenari "Identity Provider initiated", infatti, presuppongono che l'utente acceda direttamente all'autorità di certificazione d'identità, vi si autentichi e quindi selezioni il Service Provider a cui accedere scegliendo tra le possibilità offerte dal certificatore stesso (per esempio mediante una lista di link). Si vedano a tal proposito gli scenari "Identity Provider initiated" descritti in [8], per esempio le sez. 4.1.7 e 4.1.8.

Il profilo SAML "Service Provider initiated" a cui si fa riferimento nel caso dell'accesso utente via web ai servizi di front-end è il "Web Browser SSO Profile" (cfr. [8], sez. 4.1), in particolare nelle sue due versioni "POST->POST binding" (cfr. [8], sez. 4.1.1) e "Redirect->POST binding" (cfr. [8], sez. 4.1.2). Il profilo citato identifica i ruoli delle entità coinvolte e descrive la sequenza dei passi che caratterizzano le loro interazioni, parte dei quali sono mostrati nel diagramma UML che segue (si noti in particolare che sia il Local Proxy che la Profile Authority assumono il ruolo sia di "Identity Provider" che di "Service Provider" in quanto essi rivestono un ruolo di intermediario nell'interazione tra gli effettivi componenti architetturali Service Provider e Identity Provider, come illustrato nel seguito).

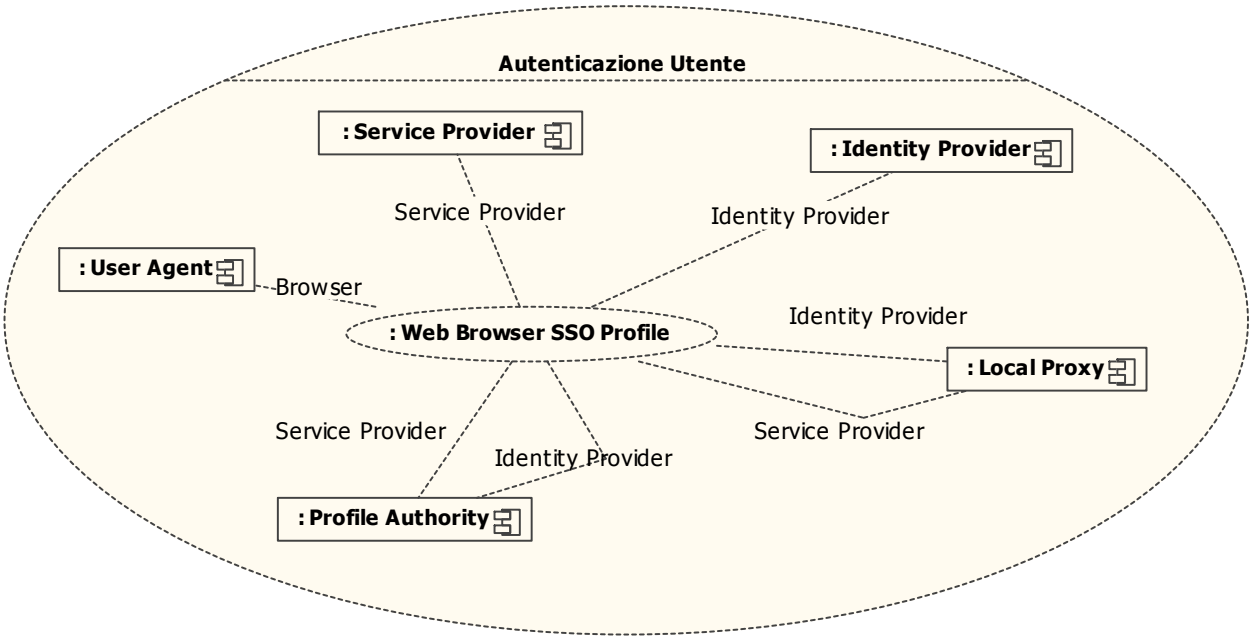

Figura 3. Mapping tra ruoli del profilo "Web Browser SSO" e componenti architetturali

Per gestire l'autenticazione dell'utente si fa ricorso al costrutto `<AuthnRequest>` e al relativo costrutto `<Response>` di risposta. In accordo con quanto definito dal profilo "Web Browser SSO", la richiesta di autenticazione viene inoltrata prima dal Service Provider al Local Proxy, poi dal Local Proxy alla Profile Authority dell'utente, e infine da quest'ultima all'Identity Provider dell'utente.

Le caratteristiche che deve avere la `<AuthnRequest>` in questo scenario sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'elemento `<Issuer>` a indicare l'entità emittente (il Service Provider, il Local Proxy o la Profile Authority, a seconda dei casi);
- può non essere presente l'elemento `<Subject>` (per esempio nel caso in cui esso non sia ancora noto al Service Provider);
- l'elemento `<NameIDPolicy>` e il relativo attributo `AllowCreate` devono segnalare all'Identity Provider che non è ammesso che l'identificativo dell'utente venga creato contestualmente alla fase di autenticazione (in altre parole, si richiede che il subject sia già registrato presso il certificatore d'identità);
- l'attributo `ForceAuthn` deve valere "false" nell'interazione tra Service Provider e Local Proxy e tra Local Proxy e Profile Authority (cioè non si richiede che né il Local Proxy né la Profile Authority autentichino direttamente l'utente) mentre deve valere "true" nell'interazione finale tra la Profile Authority e l'Identity Provider se si vogliono prevenire ulteriori redirezioni (cioè richiedere che sia proprio quell'Identity Provider ad autenticare direttamente l'utente);
- l'attributo `ProxyCount` dell'elemento `<Scoping>` deve correttamente indicare il numero di redirezioni permesse verso i certificatori di identità: deve valere almeno "2" nell'interazione tra Service Provider e Local Proxy, almeno "1" nell'interazione tra Local Proxy e Profile Authority, mentre può valere "0" nell'interazione tra Profile Authority e Identity Provider se si vuole che non siano permesse ulteriori redirezioni (vedi punto precedente);
- l'elemento `<IDPList>` dell'elemento `<Scoping>`, se presente, può contenere la lista delle entità che il Service Provider considera fidate ai fini dell'elaborazione della richiesta di autenticazione, cioè il Local Proxy (a sua volta il Local Proxy potrà indicare la Profile Authority scelta dall'utente); nell'interazione tra la Profile Authority e l'Identity Provider non occorre specificare questa informazione in quanto la scelta del certificatore d'identità deriva dal contenuto del profilo utente;
- l'elemento `<RequesterID>` dell'elemento `<Scoping>`, nel caso delle interazioni a valle di quella tra Service Provider e Local Proxy, deve indicare le entità che hanno emesso originariamente la richiesta di autenticazione e quelle che in seguito l'hanno propagata (cfr. [9], sez. 3.4.1.2 e 3.4.1.5 per le questioni legate al proxying);
- l'elemento `<Conditions>` può indicare i limiti di validità attesi dell'asserzione ricevuta in risposta;

- può essere presente l'elemento `<RequestedAuthnContext>` a indicare il contesto di autenticazione atteso (per esempio la "forza" delle credenziali richieste);
- deve essere presente l'elemento `<Signature>` apposto dal Service Provider.

Le caratteristiche che deve avere la `<Response>` che rappresenta la risposta alla richiesta di autenticazione appena descritta sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'attributo `InResponseTo`, il cui valore deve fare riferimento all'`ID` della richiesta a cui si risponde;
- deve essere presente l'elemento `<Issuer>` a indicare l'entità emittente (l'Identity Provider, la Profile Authority o il Local Proxy, a seconda dei casi);
- deve essere presente l'elemento `<Subject>` che identifica l'utente autenticato;
- deve essere presente un elemento `<Assertion>` di avvenuta autenticazione contenente un elemento `<AuthnStatement>`;
- nella `<Assertion>` di autenticazione, nell'elemento `<Conditions>` devono essere presenti i vincoli di validità dell'asserzione (per esempio `NotBefore`, `NotOnOrAfter`, `OneTimeUse`, `ProxyRestrictions`);
- nella `<Assertion>` di autenticazione, nell'elemento `<AuthnContext>` deve essere presente la descrizione del contesto di autenticazione effettivo;
- ciascuna `<Assertion>` deve recare la `<Signature>` dell'authority emittente;
- deve essere presente l'elemento `<Signature>` apposto dalla Profile Authority e dal Local Proxy, a seconda dei casi.

Nella risposta fornita dalla Profile Authority al Local Proxy sarà presente anche un attributo XML custom che indica a quale particolare profilo il Local Proxy dovrà fare riferimento nella successiva fase di recupero delle informazioni del profilo utente. Ciò vale ovviamente solo se esiste più di un profilo associato a un determinato utente. Per maggiori informazioni si veda la descrizione della `<ProfileQuery>`, sez. 4.2.5.

In tutte le risposte fornite da authority di certificazione deve essere inoltre presente l'asserzione di abilitazione (cfr. sez. 4.2.4).

Per quel che riguarda la rappresentazione della "forza" delle credenziali di autenticazione, si fa riferimento alla struttura del costrutto `<AuthnContext>` come descritto nella specifica SAML.

4.2.3. Meccanismi di autorizzazione

Il sistema INF-3 effettua le operazioni di autorizzazione sulla base di credenziali certificate contenute nel Portafoglio delle Asserzioni costruito dall'entità Local Proxy e relative agli attributi dell'utente che accede al servizio. Si osserva che l'implementazione degli specifici meccanismi di autorizzazione è di esclusiva competenza dei singoli Service Provider, i quali potranno per esempio fare ricorso allo stesso

standard SAML e in particolare ai costrutti `<AuthzDecisionQuery>` (cfr. [9], sez. 3.3.2.4) e `<AuthzDecisionStatement>` (cfr. [9], sez. 2.7.4), oppure adottare lo standard XACML [18] e le relative estensioni per il trasporto delle asserzioni di autorizzazione (cfr. per esempio [19]).

Per abilitare e supportare i meccanismi di autorizzazione si fa anzitutto riferimento al costrutto `<AttributeQuery>` e alla relativa `<Response>`. Tali costrutti servono per interrogare il Local Proxy o le Attribute Authority al fine di certificare il valore assunto da determinati attributi del profilo utente.

Le caratteristiche che deve avere in questo caso la `<AttributeQuery>` sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'elemento `<Issuer>` a indicare l'entità emittente;
- deve essere presente l'elemento `<Subject>` a indicare l'utente a cui si riferisce la richiesta di attributi;
- devono essere presenti uno o più elementi `<Attribute>`, il cui `Name` indica l'attributo di cui si vuole conoscere il valore;
- in ciascun elemento `<Attribute>` possono essere presenti elementi `<AttributeValue>` per richiedere la verifica che l'attributo abbia i valori specificati;
- deve essere presente l'elemento `<Signature>` apposto dall'entità emittente.

Le caratteristiche che deve avere la `<Response>` di risposta a una richiesta di attributi sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'attributo `InResponseTo`, il cui valore deve fare riferimento all'`ID` della richiesta a cui si risponde;
- deve essere presente l'elemento `<Issuer>` ad indicare l'entità emittente;
- deve essere presente l'elemento `<Subject>` a indicare l'utente;
- devono essere presenti una o più `<Assertion>` contenenti `<AttributeStatement>`;
- ciascuna `<Assertion>` deve avere i rispettivi elementi `<Issuer>` e `<Subject>`;
- ciascuna `<Assertion>` deve essere firmata dall'authority emittente;
- ciascuna `<Assertion>` deve contenere un elemento `<Conditions>` che ne determini i vincoli di validità;
- ciascun `<AttributeStatement>` deve contenere gli `<Attribute>` (e i relativi `<AttributeValue>`) relativi agli attributi richiesti;
- deve essere presente l'elemento `<Signature>` apposto dall'entità emittente.

Si noti che la `<Response>` ricevuta dal Local Proxy da parte di una singola Attribute Authority conterrà una `<Assertion>` che a sua volta includerà un singolo `<AttributeStatement>` che potrà contenere più `<Attribute>` e relativi `<AttributeValue>`. Tale `<Assertion>` sarà

inoltre firmata dall'Attribute Authority che l'ha emessa. In aggiunta dovrà essere presente una particolare asserzione che convalida il fatto che un dato certificatore è abilitato a emettere asserzioni di un certo tipo (cfr. sez. 4.2.4). La `<Response>` complessiva emessa dal Local Proxy e destinata al Service Provider, invece, sarà firmata dal Local Proxy stesso e potrà contenere più `<Assertion>` emesse e firmate da Attribute Authority diverse, ovvero il Portafoglio delle Asserzioni. Il Portafoglio delle Asserzioni rappresenta infatti l'insieme delle asserzioni SAML ricevute a seguito di una richiesta di autorizzazione dell'utente. Ciò costituisce l'insieme di informazioni che il Service Provider trasmetterà per esempio a un altro Service Provider nel caso di interazione in cooperazione applicativa (cfr. sez. 4.2.6).

4.2.4. *Abilitazione dei certificatori*

Le asserzioni prodotte da un certificatore (di identità o di attributo) e scambiate con altre entità possono essere considerate affidabili grazie all'apposizione della firma digitale che ne comprova l'origine. A loro volta, tuttavia, le diverse authority dovranno comprovare la propria abilitazione a firmare asserzioni, per esempio relativamente agli attributi che intendono certificare. Se così non fosse, infatti, qualunque authority potrebbe certificare qualunque tipo di attributo di qualunque utente. A tal fine, nel documento di modellazione concettuale (cfr. [2], sez. 3.2) si è scelto di fare ricorso a un garante, esterno a tutti i domini e considerato fidato da tutte le authority di tutti i domini. Tale garante prende il nome di Responsabile del Dominio di Cooperazione.

Pertanto, insieme all'asserzione contenente gli statement di attributo o di identità certificati da una certa authority deve essere inserita un'asserzione di abilitazione firmata dal garante. Tale asserzione deve indicare tutti e soli gli attributi certificabili dall'authority in questione, ed è pubblicata dal garante stesso sull'Authority Registry in modo da renderla disponibile alle entità che vogliano verificarla.

4.2.5. *Accesso alla Profile Authority*

Gli attributi in base ai quali effettuare la fase di autorizzazione sono quelli contenuti nel profilo utente gestito dalla Profile Authority. L'interrogazione della Profile Authority avviene attraverso una `<ProfileQuery>` (che è una specializzazione della `<AttributeQuery>` standard SAML), a cui segue la relativa `<Response>` in risposta.

Le caratteristiche che deve avere la `<ProfileQuery>` sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'elemento `<Issuer>` a indicare l'entità emittente, cioè il Local Proxy;
- deve essere presente l'elemento `<Subject>` a indicare l'utente a cui si riferisce il profilo;
- in aggiunta, nel caso in cui a un utente sia associato più di un profilo, deve essere presente un attributo XML custom di tipo `<xs:anyAttribute>` (in formato `Basic`, cfr. [9], sez. 8.2.3) che specifica a quale profilo in particolare si deve fare riferimento;

- devono essere presenti uno o più elementi `<Attribute>`, il cui `Name` indica l'attributo del profilo utente di cui si vuole conoscere il valore (un attributo particolare sempre presente in ogni profilo utente è l'Identity Provider);
- in ciascun elemento `<Attribute>` possono essere presenti elementi `<AttributeValue>` per caratterizzare maggiormente l'attributo su cui si vogliono reperire informazioni;
- deve essere presente l'elemento `<Signature>` apposto dall'entità emittente.

Le caratteristiche che deve avere la relativa `<Response>` sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'attributo `InResponseTo`, il cui valore deve fare riferimento all'`ID` della richiesta a cui si risponde;
- deve essere presente l'elemento `<Issuer>` a indicare la Profile Authority emittente;
- deve essere presente l'elemento `<Subject>` a indicare l'utente a cui si riferisce il profilo;
- devono essere presenti una o più `<Assertion>` contenenti `<AttributeStatement>`;
- ciascuna `<Assertion>` può contenere i rispettivi elementi `<Issuer>` e `<Subject>`;
- ciascuna `<Assertion>` deve contenere un elemento `<Conditions>` che ne determini i vincoli di validità;
- ciascun `<AttributeStatement>` deve contenere gli `<Attribute>` (e i relativi `<AttributeValue>`) relativi agli attributi del profilo utente richiesti;
- in aggiunta, ciascun elemento `<Attribute>` deve essere esteso con un attributo XML di tipo `<xs:anyAttribute>` contenente l'identificativo dell'entità in grado di certificare quello specifico attributo (in formato `EntityIdentifier`, cfr. [9], sez. 8.3.6). Si noti che, in assenza di una Attribute Authority effettiva, la Profile Authority è in grado di certificare in proprio i valori degli attributi. In tali casi l'attributo che indica l'entità in grado di certificare non sarà presente o coinciderà con l'identificativo della Profile Authority stessa;
- deve essere presente l'elemento `<Signature>` apposto dalla Profile Authority;
- deve essere presente l'asserzione di abilitazione.

4.2.6. Portafoglio delle Asserzioni

Il Portafoglio delle Asserzioni rappresenta l'insieme delle asserzioni SAML collezionate e gestite dal sistema federato a seguito di una richiesta di accesso a un servizio da parte di un utente. Il Portafoglio delle Asserzioni viene costruito dal Local Proxy, che viene interrogato dal Service Provider quando è necessaria la verifica dell'autorizzazione del soggetto richiedente il servizio, in particolare per il recupero dei valori degli attributi contenuti nel profilo utente.

Il Portafoglio delle Asserzioni è dunque un aggregato di asserzioni rilasciate da Attribute Authority, quindi contenenti statement di attributo. Come indicato in Figura 2, per rappresentarne la struttura si fa

riferimento al costrutto `<Response>` previsto dalla specifica SAML 2.0 (cfr. [9], sez. 3.3.3): tale costrutto, infatti, può contenere più `<Assertion>` (nel nostro caso, quelle che il Local Proxy ottiene dalle Attribute Authority) e dà la possibilità di specificare globalmente un `<Subject>` (cioè l'utente), un `<Issuer>` (cioè il Local Proxy stesso) e altre informazioni, di apporre una firma, ecc.

A conclusione della fase di autorizzazione dell'utente, il Portafoglio delle Asserzioni coincide esattamente con una `<Response>` SAML poiché si tratta della risposta a una `<AttributeQuery>` inoltrata dal Service Provider al Local Proxy. In seguito, la stessa struttura (a meno di elementi non più necessari, come per esempio l'attributo `InResponseTo`) sarà utilizzata dal Service Provider quando necessario, per esempio per inoltrare il Portafoglio delle Asserzioni a un servizio remoto in caso di cooperazione applicativa includendolo negli header di un messaggio SOAP (cfr. sez. 4.2.9).

4.2.7. Accesso al registry delle authority

L'Authority Registry è il componente che permette di reperire in base al nome logico di una SAML authority del sistema federato il rispettivo indirizzo specifico (URI) a cui può essere contattata (insieme ad altre informazioni relative all'authority). L'interrogazione dell'Authority Registry avviene mediante una `<RegistryQuery>` (specializzazione della `<AttributeQuery>` standard SAML), a cui segue in risposta una `<Response>`.

Le caratteristiche che deve avere la `<RegistryQuery>` sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'elemento `<Issuer>` a indicare l'entità emittente;
- deve essere presente l'elemento `<Subject>` a indicare il nome logico dell'authority;
- devono essere presenti uno o più elementi `<Attribute>`, il cui `Name` indica l'attributo dell'authority di cui si vuole conoscere il valore (per esempio "URL" a indicare l'indirizzo dell'end-point);
- deve essere presente l'elemento `<Signature>` apposto dall'entità emittente.

Le caratteristiche che deve avere la relativa `<Response>` sono le seguenti:

- deve essere presente l'attributo `ID` univoco (cfr. sez. 4.2.8);
- deve essere presente l'elemento `<Issuer>` a indicare l'Authority Registry emittente;
- deve essere presente l'elemento `<Subject>` a indicare il nome logico dell'authority;
- devono essere presenti una o più `<Assertion>` contenenti `<AttributeStatement>`;
- ciascuna `<Assertion>` può contenere un elemento `<Conditions>` che ne determini i vincoli di validità;
- ciascun `<AttributeStatement>` dovrà contenere gli `<Attribute>` (e i relativi `<AttributeValue>`) relativi agli attributi dell'authority richiesti (per esempio l'URI);
- deve essere presente l'elemento `<Signature>` apposto dall'Authority Registry.

- deve essere presente l'asserzione di abilitazione.

4.2.8. *Altre considerazioni sull'uso di SAML in INF-3*

Si descrivono di seguito ulteriori vincoli che il sistema federato di autenticazione impone sull'utilizzo dello standard SAML.

- `ID` (attributo di diversi costrutti SAML): deve essere univoco, per esempio basato su una combinazione *origine* + *timestamp*.
- `Version` (attributo di diversi costrutti SAML): deve valere sempre "2.0", coerentemente con la versione della specifica SAML adottata in ICAR.
- `<Issuer>` (elemento di diversi costrutti SAML): nell'ambito del modello INF-3 questo elemento deve obbligatoriamente essere presente e indica l'entità che ha emesso il messaggio SAML.
- `<RequestedAuthnContext>` (elemento di `<AuthnRequest>`): questo elemento può essere sfruttato nel caso in cui si voglia sempre avere la possibilità di specificare la forza delle credenziali richieste da un Service Provider a un Identity Provider.
- `<Conditions>` (elemento di `<Assertion>`): la presenza di questo elemento può essere resa obbligatoria e in particolare deve contenere gli attributi `NotBefore` e `NotOnOrAfter` che indicano opportunamente l'intervallo di validità dell'asserzione. Inoltre:
  - `<OneTimeUse>`: l'uso di questo elemento può influenzare eventuali politiche di caching adottate all'interno del modello architetturale del sistema federato di autenticazione;
  - `<ProxyRestriction>`: anche in questo caso, l'uso di questo elemento può influenzare eventuali meccanismi di propagazione di asserzioni all'interno del modello architetturale del sistema federato di autenticazione.

4.2.9. *Interazioni in cooperazione applicativa*

Nel caso generale di cooperazione applicativa inter-dominio è necessario veicolare asserzioni SAML tra entità interagenti, in particolare si deve trasmettere al Service Provider del dominio erogante le asserzioni relative all'utente che sono state raccolte nel dominio richiedente, cioè il Portafoglio delle Asserzioni. Nel caso del binding SOAP, per esempio, la specifica SAML prevede che richieste e risposte SAML siano trasportate nel body di messaggi SOAP (cfr. [12], sez. 3.2.2.1). Nel nostro caso, però, il Portafoglio delle Asserzioni non rappresenta né una vera e propria richiesta (`<AuthnRequest>` o `<AttributeQuery>`) né una risposta (`<Response>` generica), bensì un insieme di asserzioni. Per veicolare tali asserzioni nell'ambito del modello federato si definisce un header SOAP ad hoc (operazione permessa dalla specifica SAML) in cui inserire le asserzioni, e che ovviamente il ricevente deve essere in grado di interpretare.

4.2.10. Gestione delle identità degli utenti

Come illustrato in precedenza, l'informazione relativa all'utente o, in generale, al subject a cui si riferisce un'asserzione SAML viene rappresentata dall'elemento `<Subject>`. In particolare, il subject può essere identificato mediante gli elementi `<BaseID>`, `<NameID>` o `<EncryptedID>` e dai relativi attributi (salvo ulteriori meccanismi di conferma). Ciò significa che, in pratica, l'utente è identificato da una stringa di testo, più o meno qualificata.

Questo aspetto merita particolare attenzione in un contesto federato, in cui tutte le entità coinvolte avrebbero bisogno di fare riferimento a un identificativo univoco di identità dell'utente per poter svolgere più semplicemente il proprio ruolo. La specifica SAML prevede opportuni protocolli per gestire e condividere i "name identifier" dei subject (cfr. per esempio [9], sez. 3.6 e 3.8), tuttavia si giudica conveniente convergere verso una scelta che promuova il codice fiscale quale identificativo univoco dell'utente (e quindi nome del subject in tutti quei costrutti che si riferiscono all'utente) adottato da tutte le entità del sistema federato di autenticazione, gestendo localmente eventuali situazioni particolari.

4.3. Vista di dettaglio

Le sezioni che seguono dettagliano ciascuno dei componenti architetturali presentati nella vista d'insieme di sez. 4.1.

4.3.1. User Agent

Il componente User Agent rappresenta l'applicazione client usata dall'utente per accedere ai servizi di front-end offerti dal Service Provider. Si tratta di un'applicazione come un normale browser web, a cui si richiede di supportare i protocolli HTTP e HTTPS con scambio di certificati.

Lo User Agent accede alle seguenti interfacce offerte da altri componenti architetturali:

- `SP User Interface`: è l'interfaccia web mediante la quale l'utente richiede i servizi offerti dal Service Provider e accede alle relative risorse.
- `LP User Interface`: è l'interfaccia web offerta dal Local Proxy per interagire con l'utente, per esempio al fine di reperire l'informazione sul nome della Profile Authority in cui è memorizzato il profilo dell'utente.
- `IdP User Interface`: è l'interfaccia web offerta dall'Identity Provider per interagire con l'utente durante la fase di autenticazione (challenge delle credenziali).
- `PA User Interface`: è l'interfaccia web attraverso cui l'utente può interagire con la Profile Authority per creare e gestire il proprio profilo utente, per indicare il proprio l'Identity Provider di riferimento, per selezionare lo specifico profilo, ecc.

La figura che segue mostra le interfacce usate dallo User Agent e il relativo modello dei dati.

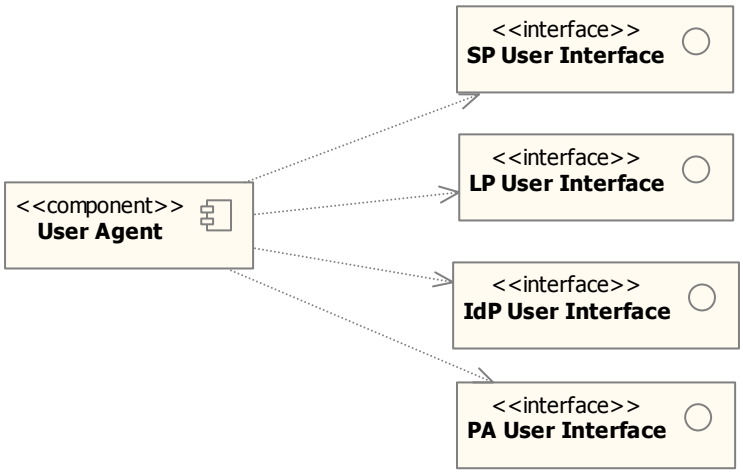

Figura 4. User Agent

4.3.2. Service Provider

Il componente Service Provider esiste a prescindere dal sistema federato di autenticazione. Il requisito che si pone su di esso è che sia SAML-aware, cioè in grado di interfacciarsi con i componenti del sistema federato di autenticazione (anzitutto il Local Proxy tramite l'interfaccia `LP Interface`) e di elaborare le informazioni con essi scambiate secondo quanto prescritto dalle specifiche SAML.

Come detto, il Service Provider espone due interfacce:

- `SP User Interface`: permette l'accesso via web tramite User Agent da parte degli utenti; l'interazione con gli utenti avviene tramite richieste e risposte HTTP o HTTPS. Gli specifici servizi offerti tramite quest'interfaccia dipendono ovviamente dal singolo Service Provider.
- `SP Interface`: permette l'interazione con altri servizi in modalità di cooperazione applicativa (attraverso il layer INF-1). Un Service Provider infatti può ricoprire non solo il ruolo di fornitore ma anche quello di fruitore in cooperazione applicativa di servizi erogati da altri Service Provider, per esempio appartenenti a domini differenti. Per questo motivo, un Service Provider interessato a prendere parte a interazioni in cooperazione applicativa deve mettere a disposizione non solo un'interfaccia utente ma anche un'opportuna interfaccia applicativa che permetta l'interazione, in qualità di erogatore di servizi, con altre entità software. Per esempio, se nell'architettura INF-3 si decidesse che questo tipo di interazione debba avvenire attraverso le tecnologie dei Web Service, l'interfaccia `SP Interface` dovrebbe essere coerente con tali tecnologie e supportarle. In INF-3 si prevede – in accordo con la "pila protocollare" descritta in [2], sez. 5.3 – che l'interazione inter-dominio sia mediata dal layer INF-1: per questo motivo, oltre a esporre l'interfaccia applicativa `SP Interface`, il Service Provider richiede che il layer INF-1 esponga la duale interfaccia `INF-1 Interface` per l'invocazione di servizi inter-dominio in qualità di fruitore (si veda a questo proposito la sez. 4.3.8). Anche l'interfaccia `INF-1 Interface` sarà coerente le tecnologie legate all'interazione applicativa, per esempio

Web Service. In ogni caso, anche gli specifici servizi offerti attraverso l'interfaccia `SP Interface` dipendono dal singolo Service Provider.

In aggiunta alla già citata interfaccia `INF-1 Interface` richiesta per l'invocazione di servizi in modalità di cooperazione applicativa, il Service Provider richiede la disponibilità della seguente interfaccia:

- `LP Interface`: questa interfaccia applicativa permette l'interrogazione del Local Proxy al fine di recuperare i valori degli attributi contenuti nel profilo utente. In questo caso il protocollo di interazione si basa sui costrutti SAML `AttributeQuery` e `Response`.

Per quel che riguarda il modello dei dati, come detto in precedenza si richiede al Service Provider di essere SAML-aware, cioè in grado di interoperare attraverso i protocolli definiti dalla specifica SAML e di gestire il relativo modello dei dati, per esempio ai fini dell'elaborazione delle informazioni contenute nei messaggi scambiati. Tra le altre, le entità che il Service Provider dovrà gestire sono:

- le richieste SAML di autenticazione (`AuthnRequest`, nella loro specializzazione interna all'architettura INF-3, cfr. sez. 4.2.1) da inoltrare ad altre entità (per esempio il componente Local Proxy, cfr. sez. 4.3.3);
- le relative risposte (`Response`, anche qui specializzate nel contesto dell'architettura INF-3);
- il `Portafoglio delle Asserzioni` restituito dal componente Local Proxy;
- eventuali costrutti SAML legati alla rappresentazione della policy di autenticazione (`AuthnContext`).

La figura che segue illustra le interfacce offerte e richieste dal Service Provider e il modello dei dati gestito per quel che riguarda le attività di autenticazione, autorizzazione e interazione con l'utente.

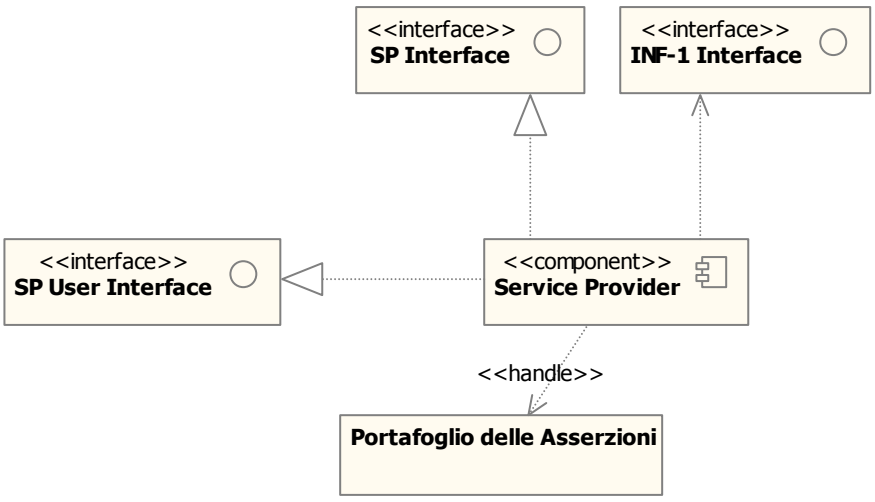

Figura 5. Service Provider

A puro titolo esemplificativo, si illustra qui la possibile architettura interna di un generico Service Provider che tenga conto dei meccanismi di autenticazione e autorizzazione proposti nel modello architetturale. Come già indicato in [2], da un punto di vista logico il Service Provider è rappresentabile come un'infrastruttura formata in particolare dai seguenti sottocomponenti:

- Gestore delle Richieste: si occupa di elaborare le richieste destinate al Service Provider e di coinvolgere opportunamente gli altri sottocomponenti per realizzare le fasi di autenticazione e autorizzazione del richiedente.
- Gestore dell'Autenticazione: si occupa della gestione del contesto di autenticazione corrente dell'utente che accede al servizio (sessione).
- Gestore delle Politiche di Autorizzazione: su richiesta del Gestore delle Richieste, si occupa di contattare il Local Proxy al fine di reperire gli attributi del profilo utente da verificare per permettere o meno l'accesso al servizio richiesto.

In aggiunta, all'"interno" del Service Provider si trovano i Servizi veri e propri, cioè le specifiche risorse a cui gli utenti desiderano accedere, tra cui possiamo distinguere:

- Servizi di Front-end: i servizi con interfaccia web destinati all'interazione con l'utente mediante browser web;
- Servizi Applicativi: servizi di back-end muniti di opportune interfacce applicative ma privi di interfaccia web.

La Figura 6 propone una possibile struttura di dettaglio del Service Provider in accordo a quanto appena descritto. Questa struttura modularizza e separa logicamente le attività di autorizzazione e autenticazione. Si può notare infatti che la gestione delle richieste, sia da parte degli utenti che applicative, viene delegata esclusivamente al sottocomponente Gestore delle Richieste. Tale sottocomponente può a sua volta coinvolgere il Gestore dell'Autenticazione o, qualora sia necessario verificare lo stato di autorizzazione dell'utente, contattare il Gestore delle Politiche di Autorizzazione. In caso di esito positivo dei controlli, verrà infine permesso all'utente autenticato e autorizzato l'accesso ai servizi richiesti.

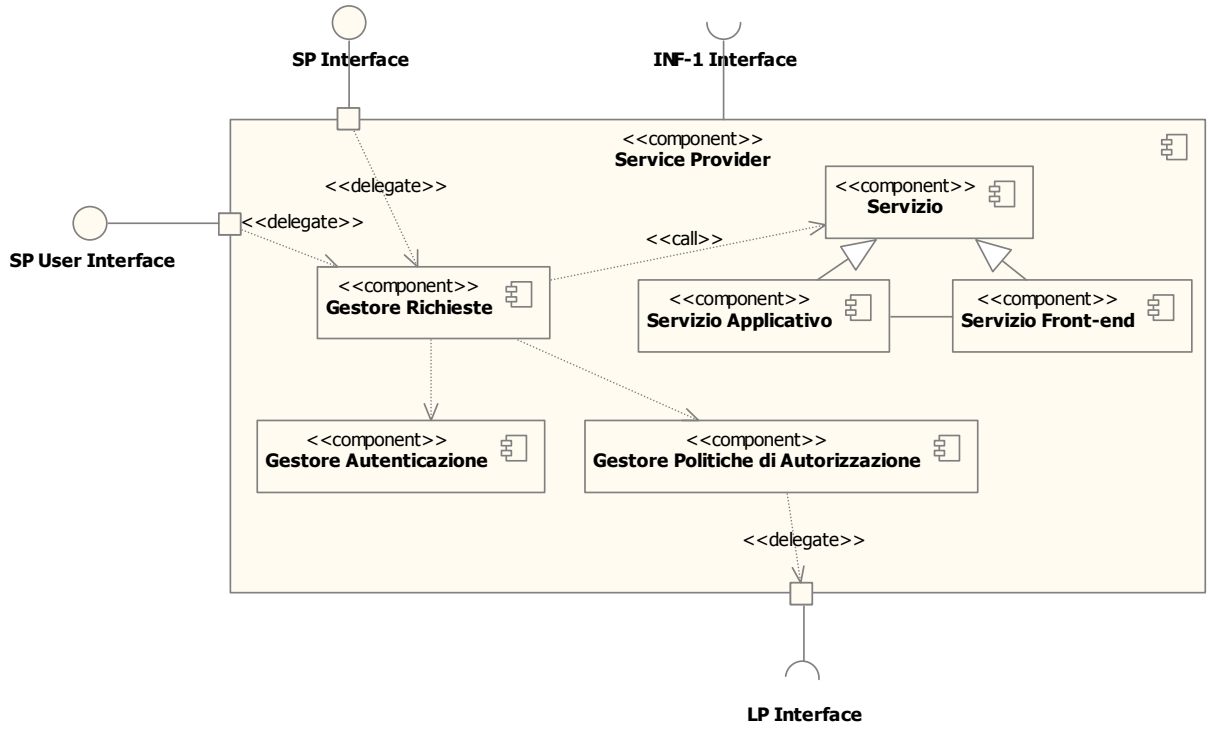

Figura 6. Service Provider (dettaglio)

A conclusione della descrizione del componente Service Provider si osserva che l'architettura interna proposta nella Figura 6 rappresenta solo una delle diverse possibilità disponibili. Nulla vieta infatti che le problematiche di autenticazione e autorizzazione siano gestite direttamente dai singoli Servizi, i quali di conseguenza si potranno rivolgere direttamente ad altri componenti architetturali come il Local Proxy.

4.3.3. Local Proxy

Il componente Local Proxy costituisce il punto di riferimento a livello di dominio per il reperimento di asserzioni SAML contenenti statement di autenticazione e/o di attributo riguardanti gli utenti. Il Local Proxy espone le seguenti interfacce:

- `LP Interface`: il Local Proxy viene interrogato dal Service Provider tramite questa interfaccia applicativa quando è necessaria la verifica dell'autorizzazione del soggetto richiedente il servizio, in particolare per il recupero dei valori degli attributi contenuti nel profilo utente. Il protocollo di interazione si basa sui costrutti `AttributeQuery` e `Response`. In particolare, il Local Proxy gestisce il `Portafoglio delle Asserzioni` che contiene l'insieme delle asserzioni collezionate a seguito di una richiesta di autorizzazione.
- `LP User Interface`: tramite questa interfaccia web il Local Proxy interagisce con il Service Provider (tramite lo User Agent dell'utente) e con l'utente:

- con il Service Provider: tramite il binding HTTP di SAML viene inoltrata la richiesta di autenticazione (`AuthnRequest`) per l'utente e in seguito ricevuta la relativa risposta (`Response`), entrambe le quali transitano sfruttando come intermediario lo User Agent dell'utente;
- con l'utente: in questo caso si è in presenza di una usuale interazione via HTTP (o HTTPS) necessaria per esempio per poter conoscere direttamente dall'utente la Profile Authority di riferimento.

Per poter svolgere le sue funzioni, il Local Proxy ha bisogno di accedere ad altri componenti architetturali, in particolare attraverso le seguenti interfacce:

- `AR Interface`: il Local Proxy ha bisogno di accedere all'Authority Registry attraverso quest'interfaccia applicativa per risolvere gli indirizzi reali di Profile Authority e Attribute Authority che devono essere consultate. L'interazione con il componente Authority Registry avviene tramite il costrutto `RegistryQuery` (una specializzazione del costrutto `AttributeQuery` previsto dal protocollo SAML, cfr. sez. 4.2.7) e relativa `Response`.
- `PA Interface`: il Local Proxy accede alle Profile Authority tramite quest'interfaccia applicativa per recuperare il profilo dell'utente che accede al servizio e poterne poi conoscere gli attributi certificati. L'interazione applicativa con il componente Profile Authority avviene tramite il costrutto `ProfileQuery` (una specializzazione del costrutto `AttributeQuery` previsto dal protocollo SAML, cfr. sez. 4.2.3) e relativa `Response`.
- `AA Interface`: il Local Proxy accede alle Attribute Authority tramite quest'interfaccia applicativa per recuperare o verificare gli attributi contenuti nel profilo dell'utente che sta accedendo al servizio. L'interazione con il componente Attribute Authority avviene tramite il costrutto `AttributeQuery` previsto dal protocollo SAML.

Il diagramma che segue mostra le interfacce offerte e richieste da un Local Proxy, e ne descrive una sua possibile struttura interna, la quale prevede la presenza di un sottocomponente avente il compito di gestire una cache delle asserzioni. Il Local Proxy infatti è il componente in cui è più vantaggioso mantenere una cache delle asserzioni riguardanti gli utenti (si vedano a tal proposito le considerazioni espresse in [2], sez. 4.6).

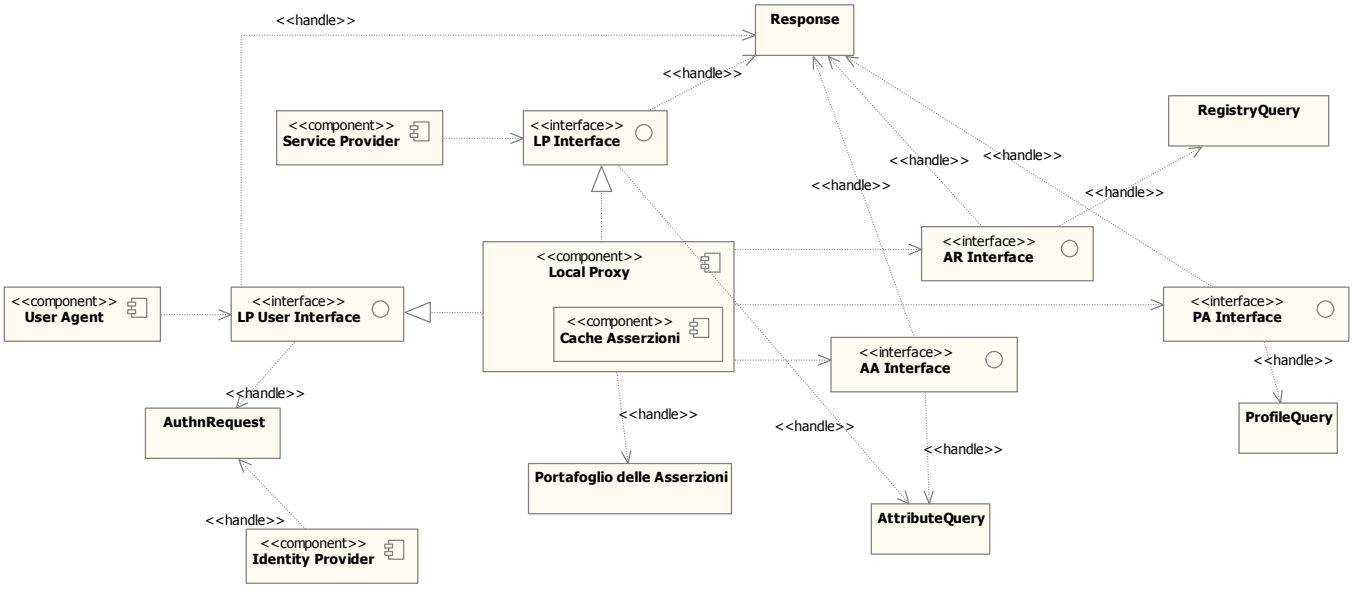

Figura 7. Local Proxy

4.3.4. Authority Registry

Il componente Authority Registry permette il reperimento di informazioni relative alle authority del sistema federato interregionale, tra cui gli specifici indirizzi (URI) a cui poterle contattare. Il componente espone la seguente interfaccia:

- `AR Interface`: si tratta di un'interfaccia applicativa di lookup, sfruttata prevalentemente dal componente Local Proxy per associare il nome logico di una authority (per esempio una Profile Authority, un Identity Provider o una Attribute Authority) con il relativo end-point. L'interazione con il Local Proxy avviene tramite il costrutto `RegistryQuery` (una specializzazione del costrutto `AttributeQuery` previsto dal protocollo SAML) e relativa `Response`. In questo modo è possibile rimanere indipendenti dalla specifica implementazione del singolo registry (UDDI, ebXML o altro). Per maggiori informazioni su questo aspetto si veda la sez. 4.2.1.

La figura che segue illustra le interfacce offerte dall'Authority Registry e il relativo modello dei dati.

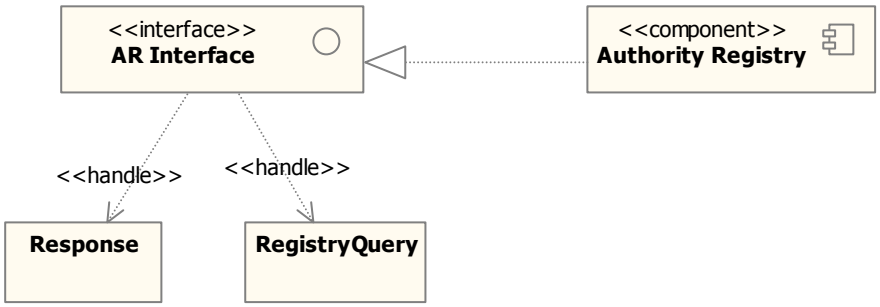

Figura 8. Authority Registry

Per maggiori informazioni sulle modalità di accesso all'Authority Registry si veda la sez. 4.2.7.

Come detto in precedenza, l'Authority Registry ha anche il compito di mantenere disponibile l'asserzione di abilitazione dei certificatori firmata dal garante (per maggiori dettagli si veda la sez. 4.2.4).

4.3.5. Identity Provider

Il componente Identity Provider è conforme all'omonima entità prevista dalle specifiche SAML, cioè rappresenta l'entità in grado di certificare l'identità di un utente a seguito di una fase di autenticazione.

Il componente espone la seguente interfaccia:

- `IdP User Interface`: è l'interfaccia utente mediante la quale l'Identity Provider può interagire con l'utente via web per effettuare la fase di "challenge" delle credenziali.

Il modello dei dati gestito fa riferimento ai costrutti SAML che riguardano i meccanismi di autenticazione (la richiesta di autenticazione `AuthnRequest` e la relativa `Response`).

La figura che segue illustra l'interfaccia offerta da un Identity Provider e il modello dei dati gestito.

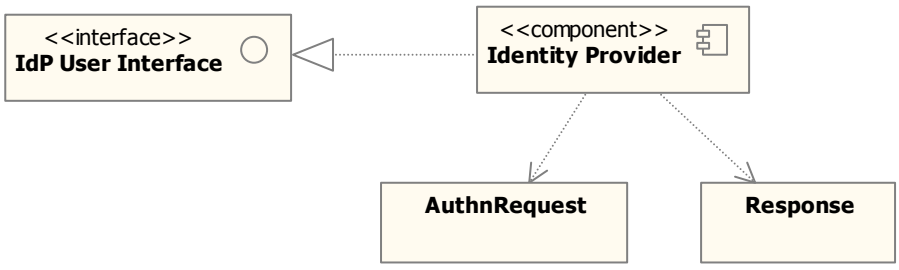

Figura 9. Identity Provider

In aggiunta a quanto mostrato nel diagramma precedente, l'Identity Provider potrà fornire agli utenti un'interfaccia web di registrazione. Tale interfaccia esula dall'ambito di competenza dell'architettura INF-3 e pertanto viene tralasciata dalla modellazione.

4.3.6. *Profile Authority*

Il componente Profile Authority memorizza e gestisce i profili degli utenti. Ciascun utente è descritto da un profilo che fa riferimento a uno o più Identity Provider in grado di certificare la sua identità. Ciascun profilo utente può contenere diversi attributi utente, ciascuno dei quali referenzia la Attribute Authority in grado di certificarlo.

Il componente espone le seguenti interfacce:

- `PA Interface`: questa interfaccia applicativa viene utilizzata dal Local Proxy quando si rivolge alla Profile Authority dell'utente per interrogare il rispettivo profilo (per esempio per conoscere i valori degli attributi del profilo utente). L'interazione con il Local Proxy avviene tramite il costrutto `ProfileQuery` (una specializzazione del costrutto `AttributeQuery` previsto dal protocollo SAML, cfr. sez. 4.2.3).
- `PA User Interface`: tramite questa interfaccia web la Profile Authority interagisce on l'utente e con il Local Proxy:
  - all'utente offre anzitutto le funzionalità di registrazione (creazione profilo) e di gestione del profilo utente (per esempio modifica o cancellazione). L'interazione in questo caso avviene attraverso il protocollo HTTP o HTTPS;
  - sempre l'utente può interagire con la Profile Authority attraverso questa interfaccia web per indicare il proprio Identity Provider o selezionare il proprio profilo di riferimento quando richiesto dalla specifica richiesta di servizio in corso;
  - tramite il binding HTTP di SAML viene inoltrata dal Local Proxy a questa interfaccia web la richiesta di autenticazione (`AuthnRequest`) per l'utente e in seguito ricevuta la relativa risposta (`Response`), entrambe le quali transitano sfruttando come intermediario lo User Agent dell'utente.

Il diagramma seguente mostra le interfacce offerte dalla Profile Authority e il modello dei dati gestito.

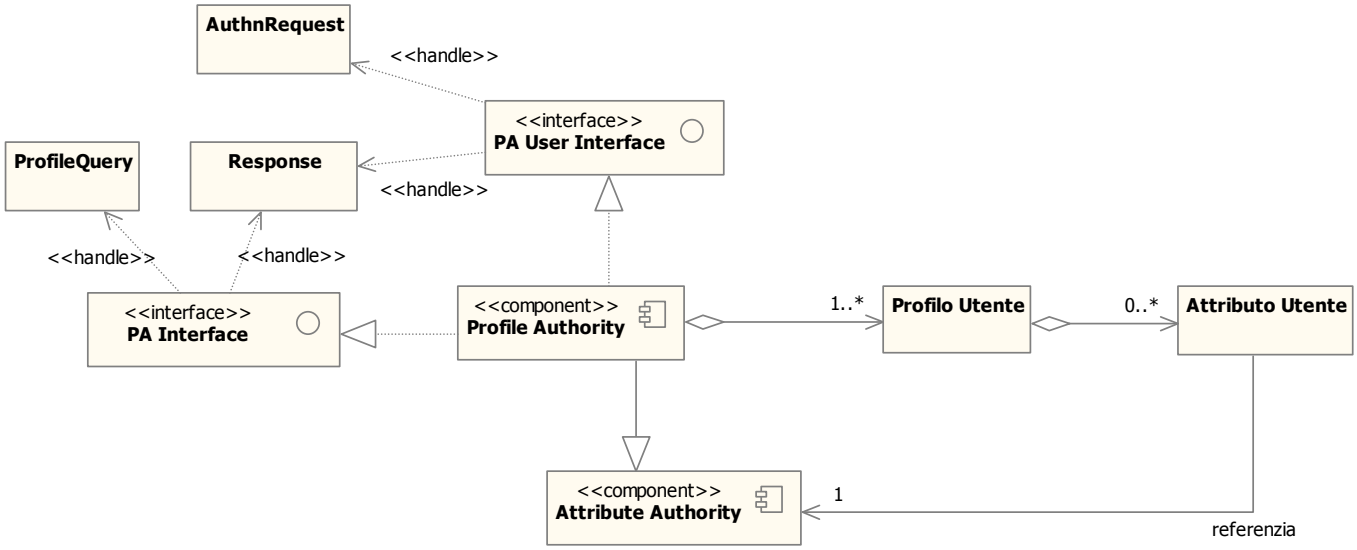

Figura 10. Profile Authority

Per maggiori informazioni sulle modalità di accesso alla Profile Authority si veda la sez. 4.2.5.

Si osserva che nel diagramma di Figura 10 la Profile Authority è modellata come una particolare Attribute Authority in quanto può svolgere il ruolo di certificatore diretto di alcuni degli attributi del profilo utente in caso di mancanza o indisponibilità di un'effettiva Attribute Authority terza (cfr. sez. 4.2.3).

Come detto in precedenza, un profilo utente è costituito da una serie di attributi. Tali attributi sono rappresentati da n-uple che contengono le seguenti informazioni:

- nome dell'attributo (per esempio "Qualifica");
- valore dell'attributo (per esempio "Ingegnere");
- nome logico dell'authority in grado di certificare l'attributo (per esempio "Ordine degli Ingegneri di Milano").

In generale, la natura dei dati memorizzati negli attributi del profilo utente sarà caratterizzata da informazioni che è necessario fornire ai Service Provider per permettere l'erogazione di determinati servizi ma che sono a conoscenza del singolo utente e non sono facilmente reperibili altrove (per esempio presso le basi di dati di amministrazioni centrali).

Affinché i dati contenuti nel profilo utente siano utilizzati coerentemente è necessario adottare uno schema di nomenclatura condiviso. A questo proposito si segnala l'approccio adottato in Internet2 e in particolare in Shibboleth (cfr. [20], sez. 5.3) per la definizione del directory schema, che si basa sull'iniziativa eduPerson [21]. Il profilo utente è comunque pensato per poter essere esteso quando necessario, per esempio per includere nella tassonomia nuovi attributi.

Per quel che riguarda la "molteplicità" delle istanze di uno stesso profilo utente, si hanno diverse possibilità a disposizione:

- profili multipli (su Profile Authority diverse): si dà cioè la possibilità all'utente di creare tanti profili diversi sulle differenti Profile Authority. Questa opzione può essere giustificata dal bisogno dell'utente di accedere a tipologie di servizi completamente differenti tra loro, per i quali le informazioni richieste a fini di autorizzazione (attributi utente) possono essere di tipo molto eterogeneo: in un caso come questo l'utente potrebbe preferire disporre di profili indipendenti tra di loro, contenenti quindi informazioni diverse (o magari parzialmente coincidenti) avendo così la possibilità di segnalare al Local Proxy quello più opportuno in fase di autenticazione. Nel caso in cui la creazione di profili diversi avvenga su Profile Authority diverse, questa soluzione ha il pregio di lasciare piena libertà all'utente ma ha lo svantaggio di poter causare una proliferazione rapida e incontrollata del numero dei profili esistenti per uno stesso utente, con il conseguente sforzo (a carico dell'utente) per mantenerli allineati.
- Profili multipli (sulla stessa Profile Authority): in questo caso si lascia all'utente la possibilità di creare più profili ma si pone il vincolo che essi esistano sulla stessa Profile Authority. Questa soluzione riduce il rischio di proliferazione del numero dei profili di uno stesso utente poiché eventuali verifiche possono essere condotte sull'unica Profile Authority di riferimento per quel dato utente. In realtà però si ipotizza implicitamente un meccanismo di comunicazione e coordinamento tra Profile Authority che impedisca strutturalmente a uno stesso utente di creare profili diversi su authority diverse.
- "Viste" dello stesso profilo: la "molteplicità" dei profili utente esistenti sulla stessa Profile Authority può essere gestita non mediante la replicazione dell'elemento, bensì tramite un meccanismo basato su "viste". In questo caso l'insieme dei dati (attributi) che costituiscono il profilo sarebbe unico, ma l'utente avrebbe la possibilità di selezionarne sottoinsiemi diversi a seconda delle esigenze, per esempio in base ai servizi di cui intende fruire.

In questo documento si assume che l'utente possa definire profili multipli sulla stessa Profile Authority.

In aggiunta a queste considerazioni, si può pensare di associare a ciascun attributo un determinato livello di visibilità sotto il controllo dell'utente, che ne limiti l'accesso a seconda dei casi (concedendo per esempio l'accesso in lettura solo a determinate entità). Inoltre, per tenere conto di specifiche esigenze legate alla privacy, è possibile pensare a opportuni meccanismi di conferma o di notifica (per esempio via e-mail) che informino l'utente ogni volta che un'entità accede a uno degli attributi del suo profilo e magari chiedano all'utente un'autorizzazione a procedere esplicita.

4.3.7. Attribute Authority

Il componente Attribute Authority ha il compito di certificare i singoli attributi che formano il profilo utente memorizzato dalle Profile Authority. Il componente espone la seguente interfaccia:

- `AA Interface`: questa interfaccia applicativa permette di reperire o verificare uno o più attributi del profilo utente. Essa viene invocata dal Local Proxy durante la fase di autorizzazione dell'utente che accede a un servizio. L'interazione con il componente Local Proxy avviene

tramite il costrutto `AttributeQuery` previsto dal protocollo SAML (cfr. sez. 4.2.3) e relativa `Response`.

La figura seguente illustra l'interfaccia offerta da un'Attribute Authority e il modello dei dati gestito.

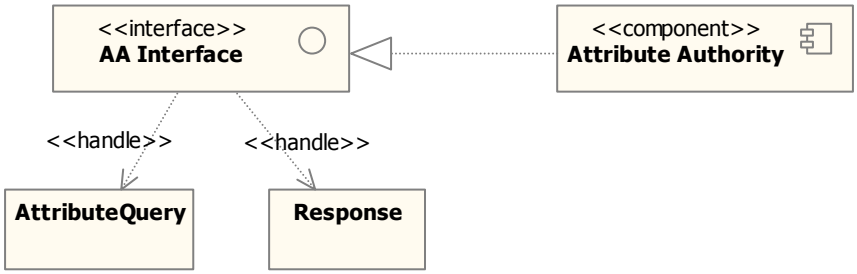

Figura 11. Attribute Authority

Valgono anche in questo caso le precisazioni espresse nella sez. 4.3.6 in merito all'adozione di una tassonomia condivisa dei nomi degli attributi.

4.3.8. Layer INF-1

Dal punto di vista dell'architettura INF-3, il layer INF-1 mette a disposizione le funzionalità di inoltro delle richieste applicative nei casi di interazione inter-dominio in cooperazione applicativa. In questa sede non è possibile dettagliare l'architettura interna di questo layer architetturale, in quanto di responsabilità di un altro intervento infrastrutturale. Pertanto, a questo riguardo si possono solo fare delle ipotesi in merito all'interfaccia attesa dallo strato INF-3. Ci si aspetta che il componente offra almeno un'interfaccia:

- `INF-1 Interface`: a questa interfaccia applicativa si rivolgeranno i componenti architetturali (anzitutto i Service Provider) quando devono fruire di servizi appartenenti a domini differenti. In tal senso, il requisito principale è che il layer INF-1 esponga opportuni metodi per inoltrare in modo trasparente e sicuro a un generico Service Provider appartenente a un altro dominio un'invocazione applicativa corredata di tutte le asserzioni SAML necessarie alla fruizione del servizio remoto (a questo proposito valgono le considerazioni già espresse in [2], sez. 5.3). A titolo puramente esemplificativo si fa l'ipotesi che il layer INF-1 esponga un'interfaccia Web service e che sia in grado di trasportare intatti gli header del messaggio SOAP contenenti le asserzioni SAML costituenti il portafoglio delle asserzioni destinato al Service Provider del dominio remoto (cfr. sez. 4.2.9).

Dualmente a quanto appena detto, ci si aspetta che il componente utilizzi la seguente interfaccia:

- `SP Interface`: per poter inoltrare al servizio destinatario le richieste di servizio in cooperazione applicativa, a sua volta il layer INF-1 contatterà i Service Provider dei domini remoti tramite l'apposita interfaccia.

La figura che segue illustra le interfacce offerte e attese relative al layer INF-1.

Figura 12. Layer INF-1

4.4.Scenari di riferimento

Al fine di illustrare come i componenti presentati finora interagiscano effettivamente tra loro, questa sezione descrive i due principali scenari di riferimento considerati nell'ambito del sistema federato di autenticazione, vale a dire l'accesso a un Service Provider da parte di un utente mediante uno User Agent e l'accesso da parte di un Service Provider a un altro Service Provider appartenente a un dominio remoto a seguito di una richiesta inoltrata da un utente mediante il proprio User Agent.

4.4.1. Accesso a servizio via web, user-initiated

Lo scenario descritto in questa sezione riguarda i meccanismi di autenticazione e autorizzazione coinvolti nell'accesso via web browser a un servizio applicativo da parte di un utente. Si noti che nel caso in esame non occorre distinguere tra dominio fruitore ed erogatore in quanto l'accesso via web scavalca tali "confini" e colloca l'utente per definizione nello stesso dominio del servizio cui accede.

Il diagramma di Figura 13 illustra lo scenario principale di interazione[1].

[1] La notazione utilizzata nei sequence diagram che seguono non è del tutto coerente con la specifica UML 2.0.

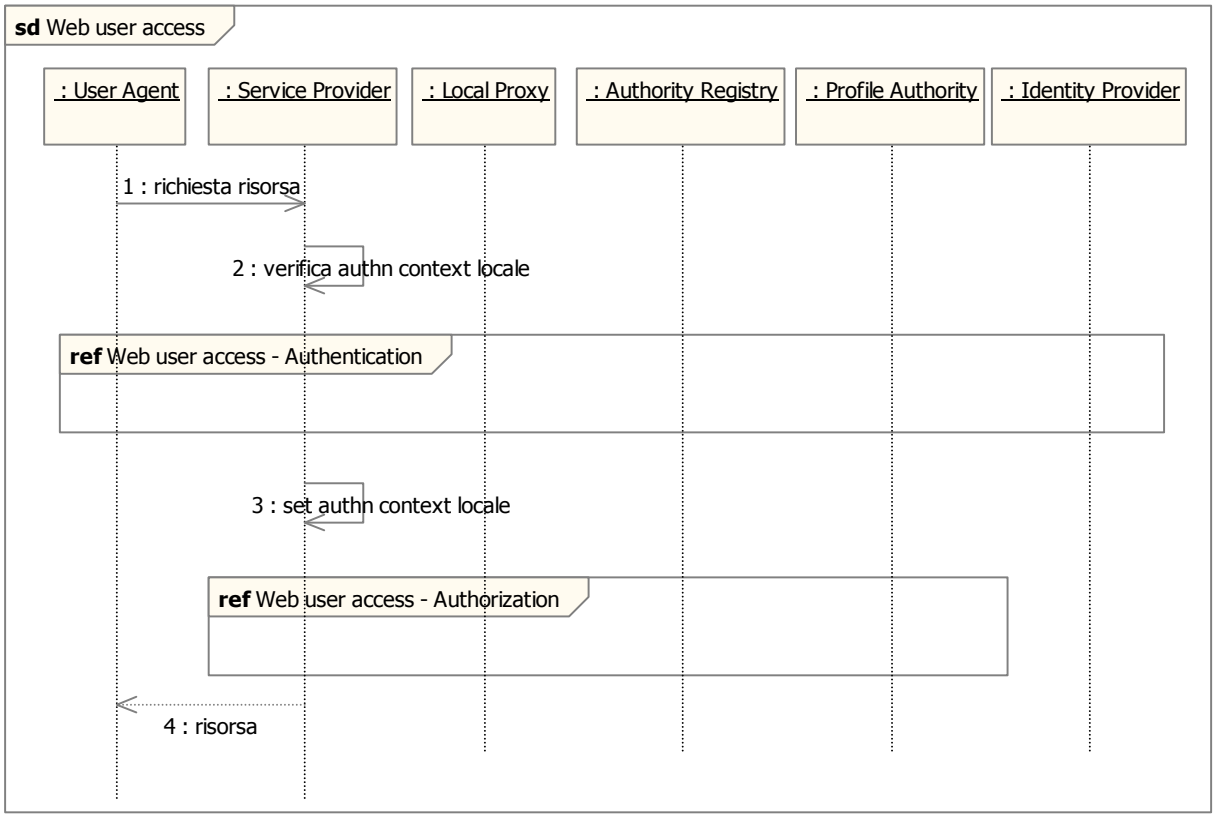

Figura 13. Scenario principale di accesso a servizio via web, user-initiated

I passi principali dello scenario illustrato sono i seguenti.

1.	Lo User Agent richiede una risorsa al Service Provider (per esempio una pagina web).
2.	Il Service Provider verifica se per quell'utente esiste già un contesto locale di autenticazione valido (per esempio una sessione gestita a livello applicativo).
	A valle di questi due passi preliminari ha luogo la fase di autenticazione, descritta in un diagramma separato (vedi più avanti).
3.	Se la fase di autenticazione va a buon fine, il Service Provider può creare un contesto locale di autenticazione e proseguire alla successiva fase di autorizzazione, anch'essa descritta in un diagramma a parte.
4.	Nel caso in cui anche la fase di autorizzazione venga superata correttamente, allo User Agent verrà fornita la risorsa richiesta (per esempio tramite redirect verso la pagina target).

Il diagramma che segue illustra in particolare la fase di autenticazione.

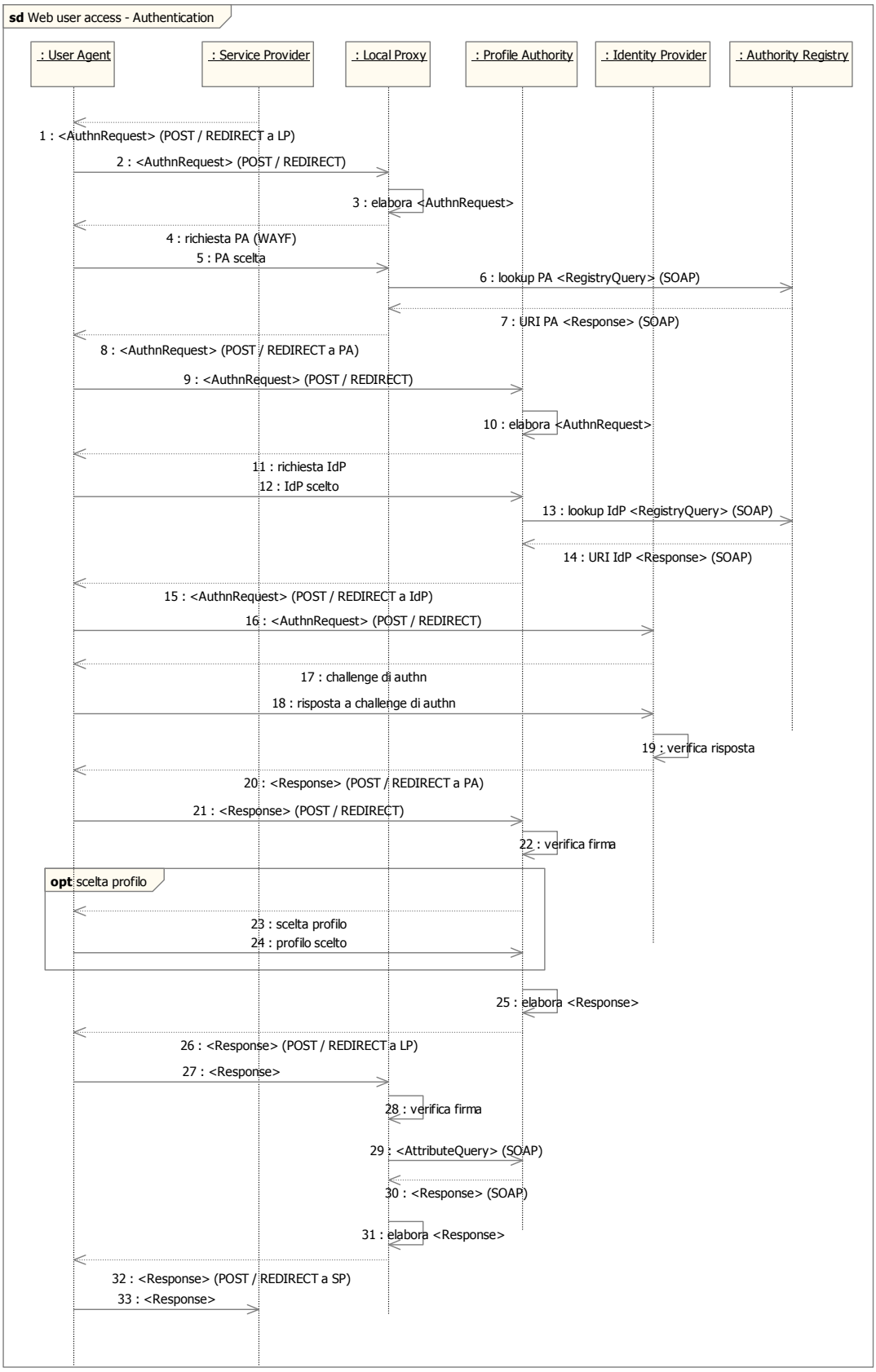

Figura 14. Scenario di autenticazione nell'accesso utente via web

I passi della fase di autenticazione sono i seguenti.

1.	A valle della verifica del contesto locale di autenticazione effettuata dal Service Provider (passo 1 dello scenario principale), nel caso l'utente non risulti autenticato il Service Provider invia allo User Agent una richiesta di autenticazione (elemento `<AuthnRequest>` della specifica SAML) da far pervenire al Local Proxy. Ciò può avvenire per esempio secondo le modalità previste dal profilo "Web Browser SSO" ("Redirect->POST" o "POST->POST").
2.	Lo User Agent inoltra la richiesta di autenticazione contattando il Local Proxy secondo la modalità adottata al passo 1.
3.	Il Local Proxy esamina la richiesta di autenticazione ricevuta dal Service Provider tramite lo User Agent.
4.	Il Local Proxy restituisce allo User Agent una form in cui richiede all'utente di indicare il nome logico della sua Profile Authority di riferimento (quest'attività viene convenzionalmente indicata con l'acronimo "WAYF", dall'espressione "Where Are You From?"); la Profile Authority di registrazione potrebbe essere dedotta dallo username qualificato dell'utente; la scelta può essere effettuata tramite una lista (es. combo box) già popolata dal Local Proxy a seguito di una opportuna query sull'Authority Registry; si noti che questa interazione non riguarda lo standard SAML.
5.	L'utente invia al Local Proxy la risposta in merito alla sua Profile Authority di riferimento (nome logico); si noti che anche questa interazione non riguarda lo standard SAML.
6.	Il Local Proxy interroga l'Authority Registry (`<RegistryQuery>`, binding SOAP) per conoscere l'URI della Profile Authority dell'utente in base al nome logico fornito.
7.	Il Registry risponde (`<Response>`, binding SOAP) restituendo al Local Proxy l'URI della Profile Authority.
8.	Il Local Proxy inoltra alla Profile Authority dell'utente la richiesta di autenticazione (`<AuthnRequest>`) attraverso lo User Agent dell'utente (modalità POST o Redirect).
9.	Lo User Agent inoltra la richiesta di autenticazione alla Profile Authority.
10.	La Profile Authority elabora la richiesta di autenticazione.
11.	La Profile Authority chiede all'utente di indicare il nome logico del proprio Identity Provider di riferimento, per esempio tramite una form restituita allo User Agent; si noti che anche questa interazione non riguarda lo standard SAML.
12.	L'utente indica il proprio Identity Provider di riferimento, restituendone alla Profile Authority il nome logico; si noti che anche questa interazione non riguarda lo standard SAML.
13.	La Profile Authority interroga l'Authority Registry (`<RegistryQuery>`, binding SOAP) per conoscere, in base al nome logico, l'URI dell'Identity Provider dell'utente.

14.	L'Authority Registry risponde (<Response>, binding SOAP) restituendo alla Profile Authority l'URI dell'Identity Provider dell'utente.
15.	La Profile Authority restituisce allo User Agent dell'utente una richiesta di autenticazione (<AuthnRequest>) diretta all'Identity Provider, analogamente a quanto illustrato al punto 1.
16.	Lo User Agent inoltra la richiesta di autenticazione contattando l'Identity Provider secondo la modalità adottata al punto 15.
17.	L'Identity Provider risponde allo User Agent iniziando la challenge di autenticazione. La tipologia di challenge può dipendere dalla <AuthnRequest> emessa dal Service Provider (per esempio possono essere espressi vincoli sulla "forza" delle credenziali richieste all'utente mediante l'elemento <RequestedAuthnContext>, cfr. sez. 4.2.2); si noti che anche questa interazione non riguarda lo standard SAML.
18.	Lo User Agent fornisce all'Identity Provider le proprie credenziali di autenticazione; si noti che anche questa interazione non riguarda lo standard SAML.
19.	L'Identity Provider verifica la risposta alla challenge di autenticazione fornita dall'utente.
20.	L'Identity Provider restituisce allo User Agent la <Response> SAML contenente l'asserzione di autenticazione dell'utente destinata alla Profile Authority.
21.	Lo User Agent inoltra alla Profile Authority l'asserzione di autenticazione emessa dall'Identity Provider.
22.	La Profile Authority verifica la correttezza della firma della <Response>.
23.	La Profile Authority chiede all'utente di selezionare il profilo da utilizzare eventualmente per le interazioni successive con il Service Provider (nel caso vi sia un solo profilo relativo all'utente richiedente questa interazione non sarà necessaria); si noti che anche questa interazione non riguarda lo standard SAML.
24.	L'utente seleziona il profilo a cui vuole che si faccia riferimento; si noti che anche questa interazione non riguarda lo standard SAML.
25.	La Profile Authority costruisce una nuova <Response>; tale risposta conterrà anche l'indicazione dello specifico profilo selezionato dall'utente.
26.	La Profile Authority restituisce allo User Agent la nuova <Response> destinata al Local Proxy.
27.	Lo User Agent inoltra la <Response> al Local Proxy.
28.	Il Local Proxy verifica la correttezza della firma della <Response>.
29.	A questo punto il Local Proxy può interagire con la Profile Authority per recuperare il contenuto del profilo utente (<AttributeQuery>) in modo da conoscere già gli attributi che contiene e le relative authority SAML in grado di certificarlo in vista dell'eventuale successiva fase di autorizzazione; l'interazione avviene attraverso il binding SOAP; la richiesta include l'indicazione del profilo utente a cui fare riferimento.

30.	La Profile Authority risponde (binding SOAP) alle richieste del Local Proxy in merito al profilo utente mediante una <Response>.
31.	Il Local Proxy costruisce una nuova <Response> che costituisce la risposta alla richiesta di autenticazione iniziale.
32.	Il Local Proxy restituisce allo User Agent la nuova <Response> destinata al Service Provider.
33.	Lo User Agent inoltra al Service Provider la <Response> di risposta alla richiesta di autenticazione iniziale.

Il diagramma che segue illustra in particolare la fase di autorizzazione.

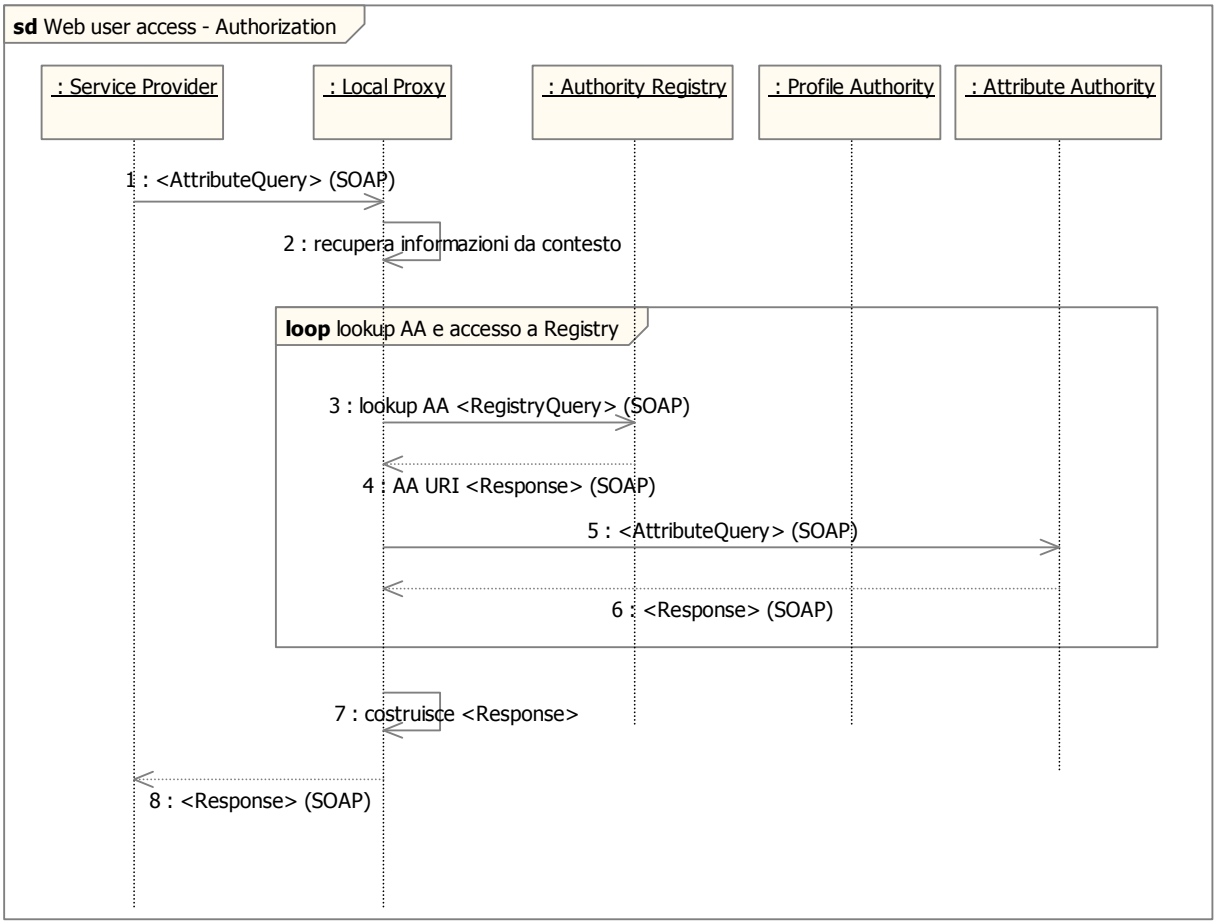

Figura 15. Scenario di autorizzazione nell'accesso utente via web

I passi della fase di autorizzazione sono i seguenti.

1.	Il Service Provider invia una `<AttributeQuery>` al Local Proxy (binding SOAP) indicando in un unico messaggio gli attributi da verificare ai fini dell'autorizzazione dell'utente.
2.	Il Local Proxy determina i dati del profilo utente in base alle informazioni già raccolte dalla Profile Authority durante la fase di autenticazione e salvate nel contesto.
3.	Il Local Proxy interroga l'Authority Registry (`<RegistryQuery>`, binding SOAP)per conoscere l'URI di ciascuna Attribute Authority.
4.	L'Authority Registry risponde (`<Response>`, binding SOAP) restituendo al Local Proxy l'URI delle Attribute Authority.
5.	Il Local Proxy invia a ciascuna Attribute Authority una `<AttributeQuery>` (binding SOAP) per ottenere la certificazione degli attributi dall'utente.
6.	Ciascuna Attribute Authority risponde restituendo una `<Response>` (binding SOAP) con le asserzioni relative agli attributi dell'utente.
7.	Il Local Proxy estrae le asserzioni e costruisce una `<Response>` globale (binding SOAP) per il Service Provider contenente le asserzioni di attributo raccolte (ciò costituisce il Portafoglio delle Asserzioni).
8.	Il Local Proxy restituisce al Service Provider la `<Response>` globale (binding SOAP) contenente le asserzioni relative a tutti gli attributi dichiarati dall'utente nel proprio profilo.

4.4.2. Accesso a servizio in cooperazione applicativa

Lo scenario in cooperazione applicativa riguarda l'interazione, a seguito di una richiesta da parte di un utente tramite User Agent, tra il Service Provider che si trova in un certo dominio (detto dominio richiedente, da cui segue la notazione SP_DR) che deve invocare un Service Provider posto in un dominio differente (detto dominio erogante, da cui segue la notazione SP_DE). L'interazione applicativa coinvolge i Layer INF-1 adibiti alla comunicazione (trasporto) interdominio.

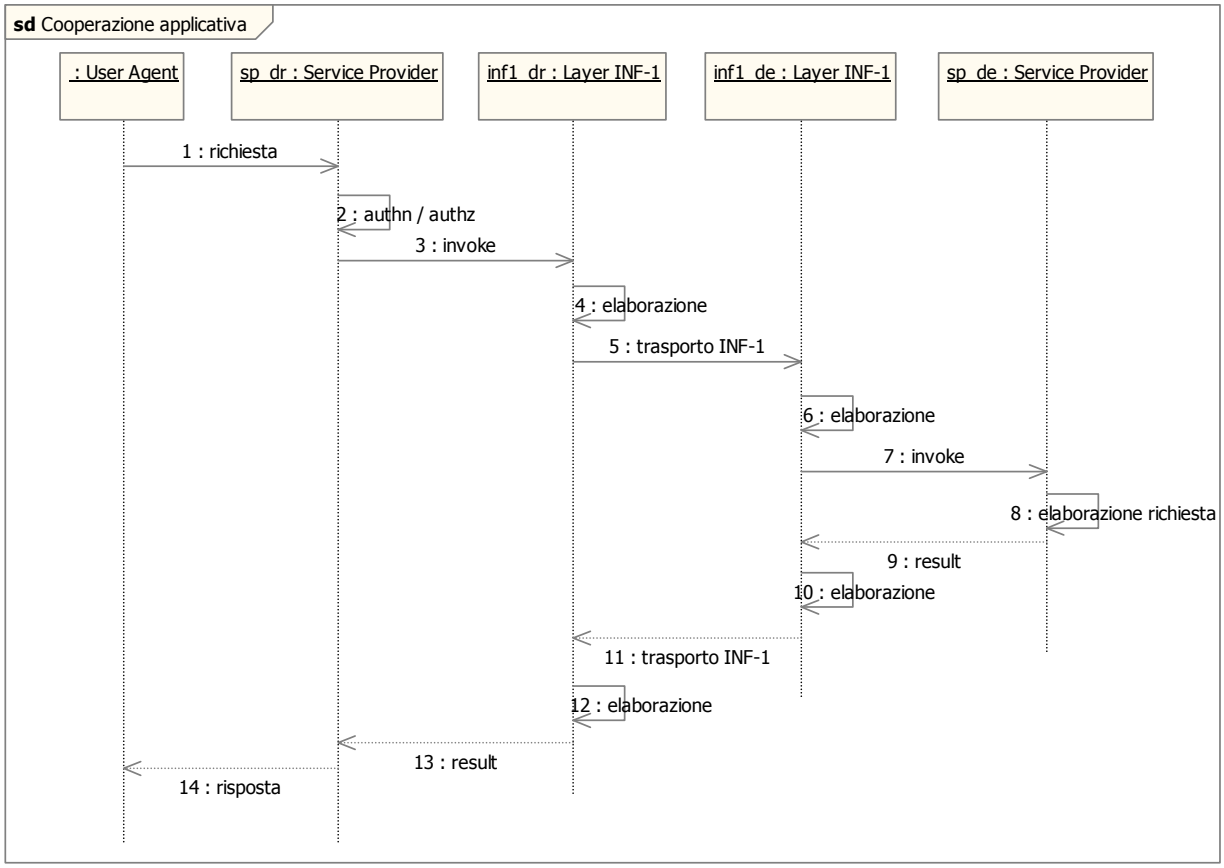

Figura 16. Scenario in cooperazione applicativa

I passi dell'interazione in cooperazione applicativa sono i seguenti.

1.	L'utente inoltra una richiesta tramite il suo User Agent.
2.	Il Service Provider svolge le opportune fasi di autenticazione/autorizzazione (non vengono qui riportati i dettagli e le altre entità coinvolte).
3.	Il Service Provider si rivolge al layer INF-1 del proprio dominio (dominio richiedente, DR) al fine di invocare il Service Provider che si trova in un altro dominio (dominio erogante, DE). Oltre all'invocazione applicativa in sé, il SP_DR passa al componente INF-1 il portafoglio di asserzioni relativo all'utente necessario alla fruizione del servizio target.
4.	Il layer INF-1_DR elabora le informazioni passate da SP_DR (per esempio crea la busta di e-government).
5.	Ha luogo la trasmissione/trasporto delle informazioni attraverso il canale di comunicazione INF-1 fino a raggiungere l'omologo layer INF-1_DE.
6.	Il layer INF-1_DE elabora le informazioni ricevute (per esempio apre ed elabora la busta di e-government).
7.	L'invocazione applicativa viene inoltrata al SP_DE.

8.	SP_DE elabora la richiesta.
9.	SP_DE invia la risposta conseguente all'invocazione applicativa rivolgendosi al layer INF-1_DE.
10.	Analogamente a quanto visto prima, il layer INF-1_DE elabora le informazioni ricevute da SP_DE.
11.	Ha luogo la trasmissione/trasporto della risposta applicativa attraverso il canale di comunicazione INF-1.
12.	Il layer INF-1_DR elabora la risposta.
13.	La risposta applicativa viene inoltrata a SP_DR.
14.	L'utente riceve infine una risposta dal Service Provider che ha invocato.

5. CONSIDERAZIONI CONCLUSIVE

Questa sezione discute gli aspetti di maggior rilievo offerti dal modello architetturale proposto ed evidenzia i rapporti tra l'ambito di competenza dell'intervento infrastrutturale INF-3 e quelli degli altri interventi infrastrutturali e dei casi di studio applicativi, nonché i legami con quanto previsto dalle specifiche SPCoop.

5.1. Valutazione del modello architetturale proposto

L'architettura proposta è in linea con i modelli di riferimento descritti nel documento di modellazione concettuale, pertanto ne eredita le caratteristiche e i vantaggi, tra i quali ricordiamo: il modesto impatto sulle entità preesistenti (come gli Identity Provider o le Attribute Authority) in termini di interfacciamento al sistema federato, il basso carico imposto alle entità in gioco e la contenuta complessità delle interazioni complete che realizzano gli scenari di riferimento.

Inoltre, il modello architetturale ha il pregio di adottare un modello dati standard e una modalità di interfacciamento omogeneo tra i componenti grazie al fatto di basarsi sulle specifiche SAML, già consolidate e normate dal punto di vista tecnologico nella quasi totalità dei loro contesti di utilizzo.

5.2. Relazioni con gli altri interventi infrastrutturali e i casi di studio applicativi ICAR

Sulla base della documentazione di progetto finora disponibile, il modello architetturale proposto non pone particolari problemi di interazione con le attività di competenza dell'intervento infrastrutturale INF-2 ("Gestione di strumenti di Service Level Agreement a livello interregionale"). In particolare, un possibile fronte di interfacciamento con questo intervento infrastrutturale riguarda l'utilizzo delle funzionalità messe a disposizione da INF-3 per disciplinare l'accesso a componenti realizzati da INF-2, quali per esempio l'indice dei livelli di servizio, il repository dei dati registrati e altre applicazioni specifiche di supporto.

Per quel che riguarda invece l'intervento infrastrutturale INF-1 ("Realizzazione dell'infrastruttura di base per l'interoperabilità e la cooperazione applicativa a livello interregionale") è da rilevare un punto critico nella comunicazione a livello di interfacciamento da e verso lo strato INF-3: nel documento si è fatta l'ipotesi che i componenti di INF-1 che gestiscono il canale di comunicazione (trasporto) tra domini offrano un'opportuna interfaccia applicativa e siano in grado di trasportare e consegnare intatto il portafoglio di asserzioni costruito a livello INF-3. Tale ipotesi va verificata con attenzione poiché da essa dipende il corretto funzionamento del sistema federato di autenticazione, in particolare per quel che riguarda lo scenario di cooperazione applicativa.

Per quel che riguarda invece i rapporti con i casi di studio applicativi, l'ipotesi è che spetti a essi la realizzazione dello strato di integrazione e adattamento dei servizi applicativi (che, si ricorda, devono essere SAML-aware) alla sottostante architettura INF-3, in particolare per quel che riguarda l'utilizzo e la gestione dei meccanismi a supporto delle fasi di autorizzazione messi a disposizione da INF-3 (cfr. per esempio [1], sez. 4).

5.3. Relazioni con le specifiche SPCoop

In questa sezione vengono illustrate le relazioni più rilevanti che legano il modello architetturale proposto nel presente documento con quanto descritto nei documenti di specifica SPCoop. L'obiettivo è illustrare quali scelte siano state in qualche modo correlate con i vincoli tecnologici, architetturali e organizzativi previsti dalle specifiche SPCoop, e in generale discutere come sia stata attuata la convergenza tra la modellazione architetturale del sistema federato interregionale di autenticazione e i requisiti del modello di cooperazione applicativa.

5.3.1. Uso di SAML 2.0

Il modello architetturale utilizza appieno lo standard SAML 2.0, previsto esplicitamente dalle specifiche SPCoop come standard di riferimento per i meccanismi di autenticazione e autorizzazione (cfr. [5], sez. 4.2.2 e 4.2.3).

5.3.2. Registry

SPCoop prevede esplicitamente un elemento architetturale chiamato Servizio di Registro SICA, il cui scopo è fornire servizi di registrazione e ricerca per la gestione di informazioni sullo stato delle seguenti risorse: soggetti organizzativi della comunità SPCoop, servizi presentati su SPCoop e domini di cooperazione (cfr. [6], sez. 3). La specifica distingue in particolare tra Registro SICA Generale e Registri SICA Secondari, ove il primo è pensato come un'istanza unica contenente la totalità delle informazioni, mentre i secondi sono istanze multiple contenenti sottoinsiemi (viste) delle informazioni contenute nel Registro SICA Generale. La specifica definisce opportuni meccanismi di accesso a tali registri e di sincronizzazione tra il Registro SICA Generale e i Secondari. La specifica sottolinea inoltre la differenza esistente in letteratura tra i termini "registry" e "repository", e fa notare che la denominazione più adatta per i Registri SICA è quella di repository.

Nel modello architetturale proposto non si pongono particolari vincoli a carico del registry/repository delle authority, salvo quello di offrire un'interfaccia basata sulle specifiche SAML. Rispettato tale vincolo, l'implementazione effettiva del registry potrà ricadere in una delle implementazioni coerenti con le specifiche dei Registri SICA basate su altre soluzioni tecnologiche.

5.3.3. Accordi di servizio e di cooperazione

Secondo le specifiche SPCoop, prima che possa avere luogo una qualsiasi interazione di servizio tra sistemi è necessario che sia stato definito un accordo esplicito che descrive le parti interagenti (non solo l'erogatore ma anche il fruitore), le funzionalità offerte, le interfacce, i requisiti di qualità e sicurezza, e altro ancora. Tale descrizione prende il nome di Accordo di servizio (o di cooperazione quando riguarda interazioni tra domini diversi). Le specifiche [7] ne descrivono le finalità, le tipologie, il ciclo di vita e i formati di descrizione (a tal proposito è bene sottolineare che ciascun accordo di servizio prevede una parte di specifica formalizzata (per esempio l'interfaccia descritta in WSDL) e una parte non formalizzata (descrizione rigorosa ma semi-formale di caratteristiche quali i livelli di qualità e sicurezza supportati).

Rispetto al modello architetturale proposto, valgono anche in questo caso i commenti espressi in merito al registry in rapporto con quanto previsto dalla specifica SPCoop, in particolare per quel che riguarda il modello dei dati adottato. In generale, il modello dati previsto dal modello architetturale si configura come un sottoinsieme modesto delle informazioni potenzialmente esprimibili da un Accordo di Servizio e comunque legato alla parte di specifica formalizzata.

5.3.4. Autorità di certificazione

SPCoop prevede esplicitamente un elemento architetturale chiamato Autorità di Certificazione (cfr. [4], sez. 3.5.4, e [5], sez. 3), avente il compito di gestire un'infrastruttura a chiave pubblica su cui basare la gestione di tutti gli aspetti crittografici. In [5] sono precisati il ruolo e le caratteristiche di detto componente, denominato PKI SPC.

Anche in ICAR INF-3 si parla di Certification Authority ma si tratta di qualcosa di diverso rispetto a quanto descritto dalle specifiche SPCoop. In questo documento, al fine di evitare possibili ambiguità, la Certification Authority è stata rinominata in Identity Provider.

5.3.5. Obiettivi di sicurezza

Come specificato in [1], l'intervento infrastrutturale INF-3 si colloca nell'ambito della realizzazione di funzioni relative alla sicurezza, in linea con gli obiettivi di sicurezza di SPCoop descritti in [5] (cioè autenticazione, autorizzazione, delega, integrità, riservatezza, non ripudiabilità, ispezione e tracciabilità, amministrazione). Sempre in [5], per ciascuno di questi requisiti di sicurezza vengono indicate delle regole a cui l'erogazione e la fruizione di un servizio applicativo devono sottostare.

Date le tecnologie adottate in INF-3, i requisiti dettati da SPCoop non presentano particolari problemi, fatto salvo forse il solo meccanismo di delega, attualmente non coperto esplicitamente (da notare che in [5] tale requisito è citato nell'elenco di sez. 4.1.1 ma non ha una sua "regola" di dettaglio nella sez. 4.2). Sono tuttavia noti in letteratura lavori che estendono opportunamente SAML per supportare la delega (cfr. per esempio [17]) e che dunque possono essere utilizzati per integrare tale meccanismo nel funzionamento dell'infrastruttura INF-3.

5.3.6. *Marcatura temporale*

SPCoop prevede esplicitamente l'esistenza di un servizio di marcatura temporale che realizza un meccanismo fidato in grado di attestare in modo certo l'esistenza di un documento informatico a una certa data e a una certa ora (cfr. [5], sez. 5.1.3).

Tale servizio potrebbe essere utilizzato nel sistema federato di cooperazione, per esempio al fine di specificare nelle asserzioni e in generale nei messaggi SAML l'attributo relativo all'istante temporale di emissione (`IssueInstant`).

6. EVOLUZIONE DEL MODELLO

Il modello architetturale sin qui descritto costituisce il nucleo base del sistema federato interregionale di autenticazione, ed è pensato per poter evolvere in direzioni diverse a seconda delle esigenze: nuovi contesti d'uso, nuovi scenari di interazione, maggiore complessità tecnologica, aggiornamento delle normative, ecc. L'obiettivo delle sezioni che seguono è di illustrare e discutere alcune possibilità di estensione del modello emerse durante la fase di modellazione architetturale sin qui condotta.

6.1. Evoluzione dei meccanismi di autorizzazione

In questa sezione si presenta una versione dei meccanismi di autorizzazione estesa facendo ricorso allo standard XACML.

6.1.1. Uso di XACML

Come detto in precedenza, spetta al singolo Service Provider la gestione delle proprie politiche di autorizzazione alle risorse e ai servizi offerti. Tenendo presente l'utilizzo di SAML quale standard per la rappresentazione e lo scambio di asserzioni, appare naturale sfruttare una delle seguenti possibilità per la gestione dei meccanismi di autorizzazione:

- utilizzare dei costrutti SAML dedicati alla gestione dell'autorizzazione, in particolare `<AuthzDecisionQuery>` (cfr. [9], sez. 3.3.2.4) e `<AuthzDecisionStatement>` (cfr. [9], sez. 2.7.4);
- adottare lo standard XACML [18] e le relative estensioni per il trasporto delle asserzioni contenenti statement di autorizzazione, in particolare il SAML Profile of XACML [19], nonché per la rappresentazione delle policy di autorizzazione.

Considerato che i costrutti SAML 2.0 relativi alla gestione dell'autorizzazione sono definiti "frozen" (cfr. per esempio [9], sez. 2.4.7 e 3.3.2.4), si ritiene vantaggioso accogliere il suggerimento dato dalla specifica SAML 2.0 stessa e fare evolvere il modello verso l'utilizzo di XACML quale standard per la rappresentazione e la gestione delle politiche di autorizzazione.

Il primo passo in questa direzione consiste nel trovare una corrispondenza tra le entità del modello architetturale e i ruoli previsti dalla specifica XACML [18], in particolare quelli di Policy Information Point (PIP), Policy Decision Point (PDP), Policy Enforcement Point (PEP), Policy Administration Point (PAP) e Context Handler.

Poiché il Portafoglio delle Asserzioni viene costruito dal Local Proxy interagendo con diverse Attribute Authority, l'entità Local Proxy svolge proprio il ruolo di PIP definito in XACML.

L'entità responsabile di effettuare il policy enforcement per la fase di autorizzazione e permettere o negare all'utente l'accesso ai servizi richiesti è il Service Provider stesso, per esempio tramite un suo sottocomponente dedicato alla gestione delle richieste provenienti dagli utenti. Tale entità svolge le funzionalità corrispondenti al ruolo di PEP.

Il Service Provider svolge anche il ruolo di PAP, cioè definisce le politiche di accesso ai servizi e le rende accessibili da parte del PDP (per esempio archiviandole in un repository delle policy).

Il meccanismo vero e proprio di decisione è responsabilità dell'entità Gestore delle Politiche di Autorizzazione. Tale entità svolge le funzionalità corrispondenti ai ruoli di PDP e di Context Handler definiti dalla specifica XACML.

Il diagramma che segue mostra il mapping di alto livello tra i ruoli XACML e i componenti architetturali dell'infrastruttura INF-3.

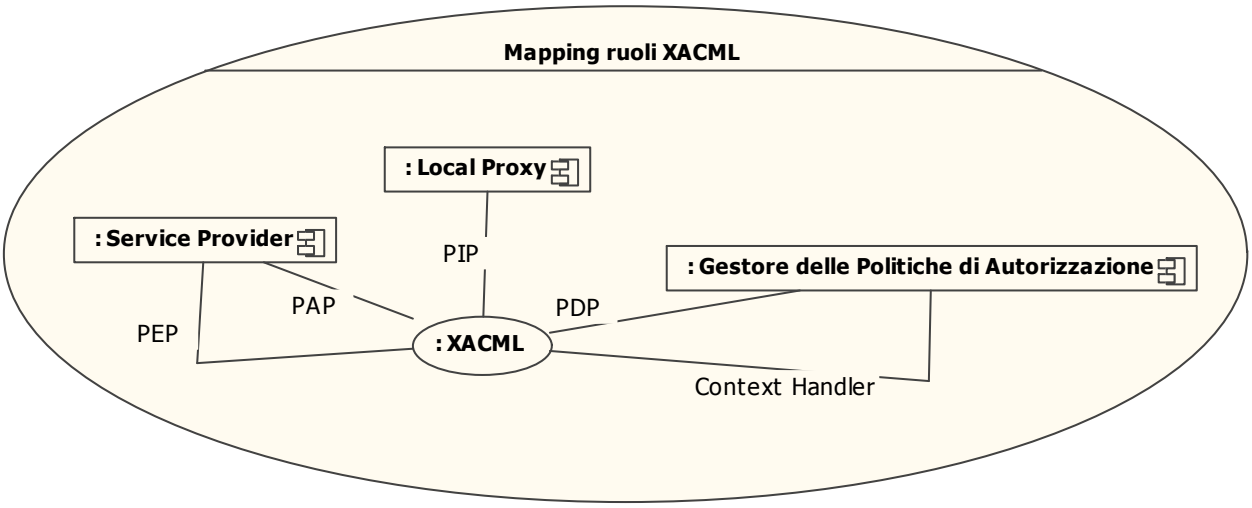

Figura 17. Mapping tra ruoli XACML e componenti architetturali INF-3

Per maggiori informazioni sullo standard XACML si vedano [18] e [19].

La figura che segue completa il quadro già illustrato in Figura 2 mostrando anche alcuni degli elementi XACML considerati nel contesto di possibili evoluzioni del modello architetturale di INF-3, che verranno illustrate nel seguito.

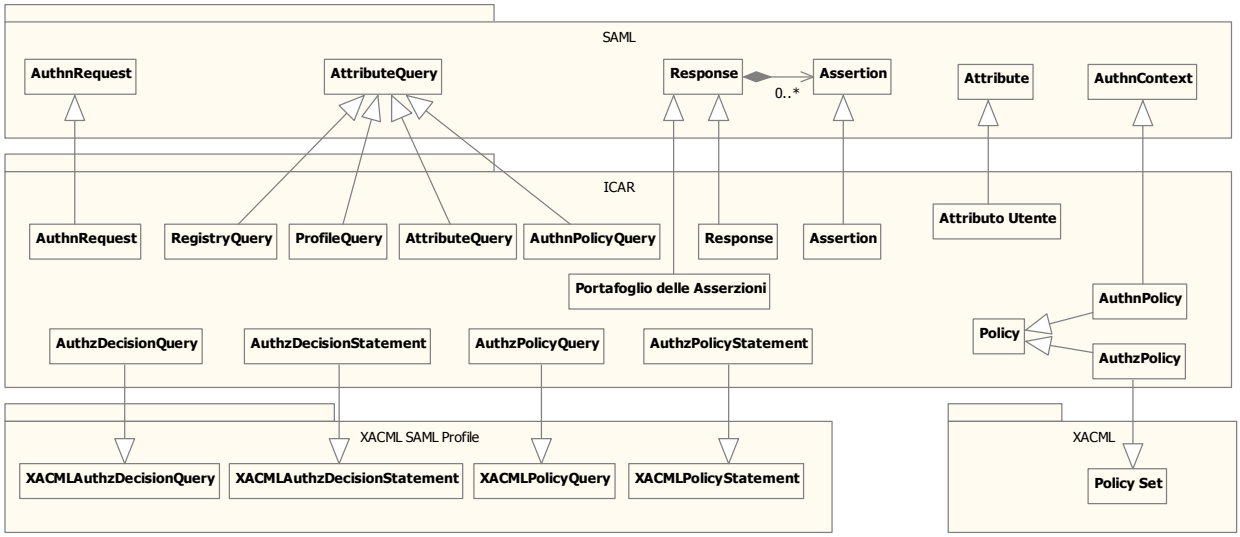

Figura 18. Entità SAML e XACML considerate nel contesto dell'architettura INF-3

Sulla base di quanto visto in merito al mapping tra ruoli XACML e componenti architetturali, le sezioni che seguono dettagliano il modello sin qui proposto ampliando e approfondendo la descrizione del già citato Gestore delle Politiche di Autorizzazione e introducendo un nuovo componente chiamato Policy Repository.

6.1.2. Gestore delle Politiche di Autorizzazione

Il componente Gestore delle Politiche di Autorizzazione ha il compito di verificare lo stato di autorizzazione dell'utente richiedente. Questo componente può essere visto come un sottocomponente del Service Provider (come già illustrato nella sez. 4.3.2) oppure come un componente che esiste a sé stante. In quest'ultimo caso è il Gestore delle Politiche di Autorizzazione a contattare il Local Proxy su richiesta del Service Provider al fine di reperire gli attributi del profilo utente da verificare per stabilire se permettere o meno l'accesso al servizio richiesto.

Pertanto, nell'ipotesi di considerare il Gestore delle Politiche di Autorizzazione come entità a sé stante rispetto al Service Provider, il componente offre l'interfaccia seguente:

- `GPA Interface`: tramite quest'interfaccia applicativa il Service Provider può rivolgersi al Gestore delle Politiche di Autorizzazione chiedendo di verificare lo stato di autorizzazione di un determinato utente che chiede di accedere a un certo servizio. Affinché sia possibile effettuare quest'operazione, il servizio dovrà essere stato associato a un'opportuna policy di accesso descritta in precedenza e poi memorizzata nel componente Policy Repository (si veda la sez. 6.1.3). Le interazioni con il Service Provider avvengono sulla base del protocollo previsto dal profilo SAML per XACML, in particolare attraverso i costrutti `AuthzDecisionQuery` e `AuthzDecisionStatement`).

Analogamente, il Gestore delle Politiche di Autorizzazione utilizza le funzionalità offerte da altri componenti architetturali attraverso le seguenti interfacce:

- `LP Interface`: questa interfaccia applicativa permette l'interrogazione del Local Proxy al fine di recuperare i valori degli attributi contenuti nel profilo utente. In questo caso il protocollo di interazione si basa sui costrutti SAML `AttributeQuery` e `Response`.
- `PR Interface`: tramite questa interfaccia applicativa il Gestore delle Politiche di Autorizzazione può recuperare dal componente Policy Repository le descrizioni delle policy di accesso ai servizi forniti dal Service Provider. L'interazione avviene coerentemente ai costrutti specificati dal profilo SAML per XACML, in particolare `PolicyQuery` e `PolicyStatement`.

La figura che segue illustra le interfacce offerte e richieste dal Gestore delle Politiche di Autorizzazione e il relativo modello dei dati gestito.

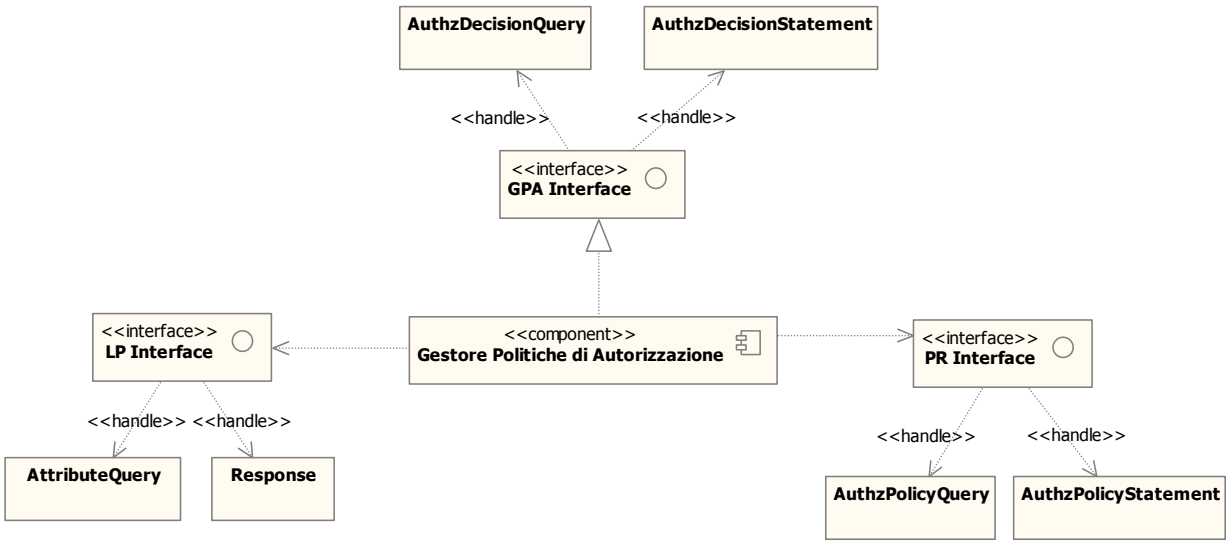

Figura 19. Gestore delle Politiche di Autorizzazione

6.1.3. *Policy Repository*

Dall'analisi svolta nella sez. 6.1.1 emerge l'esigenza di affiancare al Gestore delle Politiche di Autorizzazione un repository delle policy di autorizzazione. Il Policy Repository è il componente a cui è affidato il compito di gestire la memorizzazione e il reperimento delle politiche di accesso ai servizi offerti dai Service Provider. Grazie a questo componente, ciascun Service Provider che vuole sfruttare il supporto ai meccanismi di autorizzazione offerti dall'infrastruttura INF-3 può descrivere mediante XACML le specifiche policy che determinano l'accesso ai propri servizi resi disponibili agli utenti, e poi memorizzarle in un repository al fine di renderle consultabili da parte del Gestore delle Politiche di Autorizzazione.

Il Policy Repository espone la seguente interfaccia:

- `PR Interface`: grazie a questa interfaccia applicativa i Service Provider possono archiviare le politiche di accesso ai propri servizi. L'interazione con i Service Provider avviene in accordo alle specifiche del profilo SAML per XACML, in particolare utilizzando i costrutti `AuthzPolicyStatement`. Inoltre, questa interfaccia applicativa rende possibile il reperimento delle specifiche policy di accesso del servizio (sia di autenticazione che di autorizzazione. L'interazione in questo caso può avvenire in due modi diversi:
  - attraverso il costrutto `AuthnPolicyQuery` e relativa `Response` nel caso si voglia reperire la policy di autenticazione del servizio (`AuthnPolicy`, rappresentata dalla descrizione di un contesto di autenticazione);
  - in linea con i costrutti specificati dal profilo SAML per XACML, in particolare `AuthzPolicyQuery` e `AuthzPolicyStatement`, in caso di reperimento della policy di autorizzazione (`AuthzPolicy`).

La figura che segue mostra le interfacce offerte da un Policy Repository e il relativo modello dei dati.

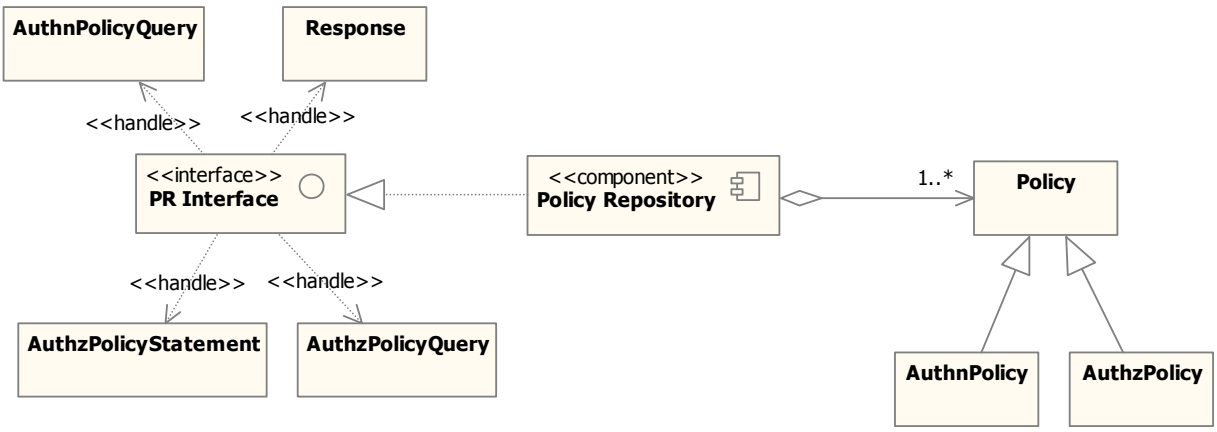

Figura 20. Policy Repository

Come detto, le policy di accesso a un servizio sono essenzialmente di due tipi:

- policy di autenticazione (`AuthnPolicy`): rappresentano la "forza" delle credenziali richieste ai fini dell'autenticazione dell'utente e possono esprimere eventuali altri vincoli; si mappano sul costrutto `AuthnContext` di SAML;
- policy di autorizzazione (`AuthzPolicy`): rappresentano le politiche di autorizzazione al servizio in termini di condizioni sui valori assunti dagli attributi del profilo dell'utente richiedente; si mappano sul costrutto `Policy` di XACML.

Il diagramma che segue mostra la vista d'insieme dell'architettura del sistema federato INF-3 a valle della revisione del ruolo del componente Gestore delle Politiche di Autorizzazione e dell'introduzione del componente Policy Repository.

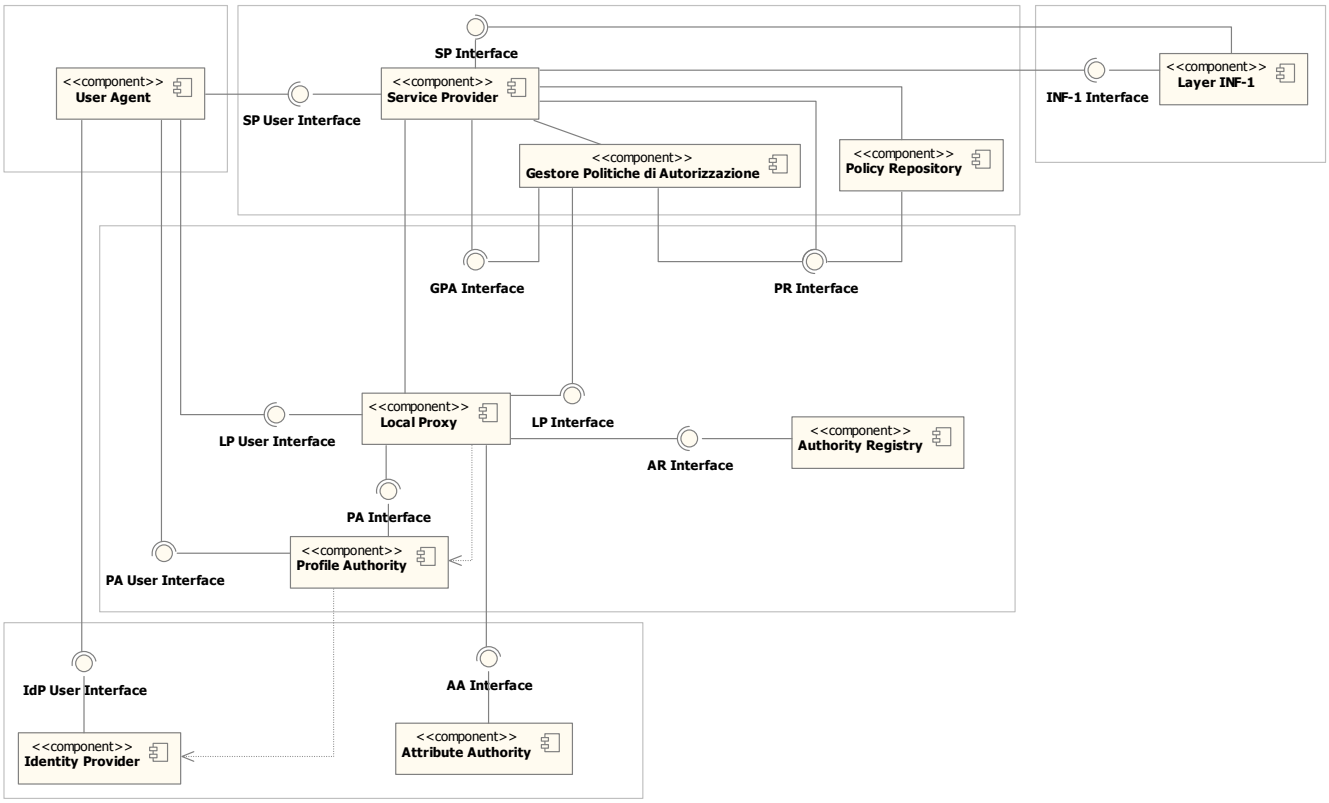

Figura 21. Nuova vista architetturale d'insieme del sistema federato di autenticazione

6.2. Evoluzione degli scenari di riferimento

Alla luce delle possibili estensioni del modello architetturale appena descritte, e in particolare quelle relative ai componenti Gestore delle Politiche di Autorizzazione e Policy Repository, è interessante mostrare l'evoluzione dei due principali scenari di riferimento considerati nell'ambito del sistema federato di autenticazione e già descritti nella sez. 4.4, vale a dire l'accesso a un Service Provider da parte di un utente mediante il suo User Agent e l'accesso da parte di un Service Provider a un altro Service Provider appartenente a un dominio remoto a seguito di una richiesta inoltrata da un utente mediante il suo User Agent.

6.2.1. Accesso utente a un servizio di front-end

Lo scenario descritto in questa sezione riguarda sempre i meccanismi di autenticazione e autorizzazione coinvolti nell'accesso via web browser a un servizio applicativo da parte di un utente. Si noti che nel caso in esame non occorre distinguere tra dominio fruitore ed erogatore in quanto l'accesso via web scavalca tali "confini" e colloca l'utente per definizione nello stesso dominio del servizio cui accede. Si noti inoltre che vengono qui tralasciati dettagli relativi alla gestione locale di sessioni di autenticazione utente da parte del fornitore del servizio.

Il diagramma che segue illustra in particolare la fase iniziale di autenticazione.

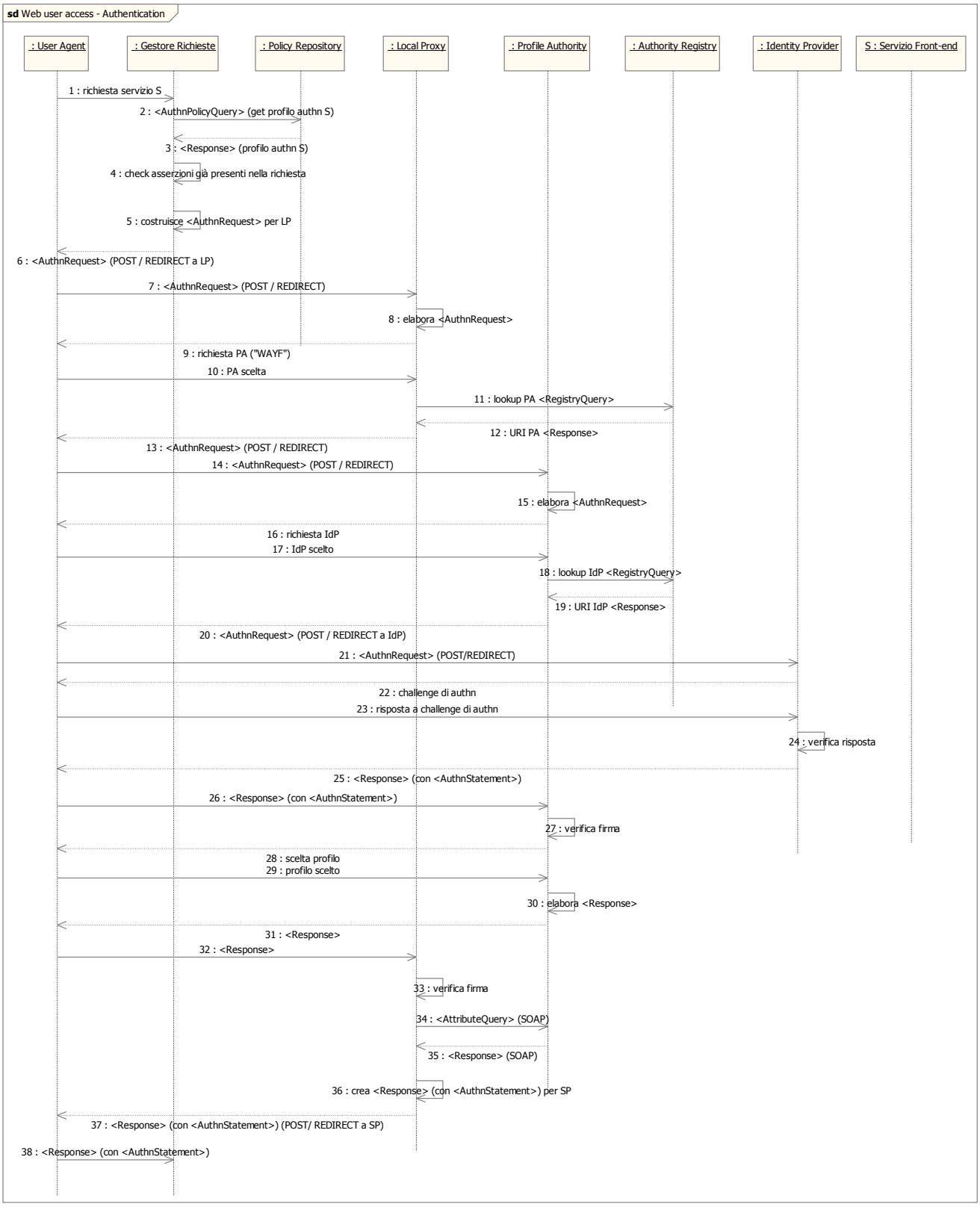

Figura 22. Scenario di autenticazione nell'accesso utente via web

I passi della fase di autenticazione sono i seguenti.

1. Lo User Agent richiede un servizio di front-end S (per esempio una pagina web) contattando il Gestore delle Richieste di un determinato Service Provider.
2. Il Gestore delle Richieste, in base alla richiesta inoltrata dall'utente, contatta il Policy Repository per ottenere il profilo di autenticazione del servizio S, contenente la descrizione del contesto di autenticazione richiesto ("forza" delle credenziali).
3. Il Policy Repository restituisce il profilo di autenticazione del servizio S.
4. Il Gestore delle Richieste controlla se nella richiesta dell'utente sono già presenti delle asserzioni di autenticazione (per esempio, l'utente potrebbe essersi già autenticato presso un Identity Provider e di lì aver poi contattato il fornitore del servizio, dando così luogo a uno scenario "IdP-initiated"). Nel seguito dello scenario si fa l'ipotesi che l'utente non sia già autenticato.
5. Poiché l'utente non è ancora autenticato, il Gestore delle Richieste costruisce una richiesta di autenticazione utente, da far pervenire al Local Proxy.
6. Il Gestore delle Richieste invia allo User Agent una richiesta di autenticazione (<AuthnRequest>) che definisce i requisiti di autenticazione richiesti. Ciò può avvenire secondo le modalità relative al profilo SAML denominato "Web Browser SSO".
7. Lo User Agent inoltra la richiesta di autenticazione contattando il Local Proxy secondo la modalità adottata al passo 6.
8. Il Local Proxy esamina la richiesta di autenticazione ricevuta dal Gestore delle Richieste tramite lo User Agent.
9. Il Local Proxy restituisce allo User Agent una form in cui chiede all'utente di indicare il nome logico della sua Profile Authority di riferimento (quest'attività viene convenzionalmente indicata con l'acronimo "WAYF", dall'espressione "Where Are You From?"); la Profile Authority di registrazione può essere dedotta dallo username qualificato dell'utente; la scelta da parte dell'utente può essere facilitata proponendo una lista (per esempio una combo box) già popolata dal Local Proxy a seguito di una opportuna query preliminare sull'Authority Registry (non mostrata nel diagramma).
10. L'utente invia al Local Proxy la risposta (HTTP POST) in merito al nome della sua Profile Authority di riferimento.
11. Il Local Proxy interroga l'Authority Registry (<RegistryQuery>) per conoscere l'URI della Profile Authority dell'utente in base al nome logico.
12. L'Authority Registry risponde restituendo al Local Proxy l'URI della Profile Authority (<Response>).
13. Il Local Proxy invia allo User Agent una richiesta di autenticazione (<AuthnRequest>) diretta alla Profile Authority.
14. Lo User Agent inoltra la richiesta di autenticazione alla Profile Authority.
15. La Profile Authority elabora la richiesta di autenticazione.

16. La Profile Authority chiede all'utente di indicare il nome logico del proprio Identity Provider di riferimento, per esempio tramite una form restituita allo User Agent.
17. L'utente indica il proprio Identity Provider di riferimento, restituendone alla Profile Authority il nome logico.
18. La Profile Authority interroga l'Authority Registry (<RegistryQuery>) per conoscere, in base al nome logico, l'URI dell'Identity Provider dell'utente.
19. L'Authority Registry risponde (<Response>) restituendo alla Profile Authority l'URI dell'Identity Provider dell'utente.
20. La Profile Authority invia allo User Agent una richiesta di autenticazione (<AuthnRequest>) diretta all'Identity Provider.
21. Lo User Agent inoltra la richiesta di autenticazione (<AuthnRequest>) contattando l'Identity Provider.
22. L'Identity Provider risponde allo User Agent iniziando con l'utente la fase di challenge di autenticazione. La tipologia di challenge può dipendere dalla <AuthnRequest> emessa dal Service Provider (per esempio possono essere espressi vincoli sulla "forza" delle credenziali richieste all'utente mediante l'elemento <RequestedAuthnContext>).
23. L'utente fornisce all'Identity Provider tramite lo User Agent le proprie credenziali di autenticazione.
24. L'Identity Provider verifica la risposta alla challenge di autenticazione fornita dall'utente.
25. L'Identity Provider restituisce tramite lo User Agent una form HTML la risposta (<Response>) contenente l'asserzione di autenticazione dell'utente (<AuthnStatement>) destinata alla Profile Authority.
26. Lo User Agent inoltra alla Profile Authority la risposta contenente lo statement di autenticazione emesso dall'Identity Provider.
27. La Profile Authority verifica la correttezza della firma.
28. La Profile Authority chiede all'utente di selezionare il profilo da utilizzare eventualmente per le interazioni successive con il Service Provider (nel caso vi sia un solo profilo relativo all'utente richiedente questa interazione non sarà necessaria); si noti che anche questa interazione non riguarda lo standard SAML.
29. L'utente seleziona il profilo a cui vuole che si faccia riferimento; si noti che anche questa interazione non riguarda lo standard SAML.
30. La Profile Authority estrae l'asserzione di autenticazione inoltrata dallo User Agent e costruisce una nuova risposta (<Response>) di autenticazione per il Local Proxy.
31. La Profile Authority restituisce allo User Agent la risposta (<Response>) di autenticazione destinata al Local Proxy.
32. Lo User Agent inoltra la risposta (<Response>) di autenticazione al Local Proxy.
33. Il Local Proxy verifica la correttezza della firma.

34. A questo punto il Local Proxy può interagire con la Profile Authority per recuperare il contenuto del profilo utente (`<AttributeQuery>`) in modo da conoscere già gli attributi che contiene e le relative authority SAML in grado di certificarlo in vista dell'eventuale successiva fase di autorizzazione; l'interazione avviene attraverso il binding SOAP; la richiesta include l'indicazione del profilo utente a cui fare riferimento.
35. La Profile Authority risponde (binding SOAP) alle richieste del Local Proxy in merito al profilo utente.
36. Il Local Proxy costruisce una nuova risposta destinata al Gestore delle Richieste.
37. Il Local Proxy restituisce allo User Agent la nuova risposta destinata al Gestore delle Richieste.
38. Lo User Agent inoltra al Gestore delle richieste la risposta alla richiesta iniziale di autenticazione.

A questo punto l'utente è stato correttamente autenticato.

Il diagramma che segue illustra la fase di autorizzazione, successiva a quella di autenticazione appena descritta.

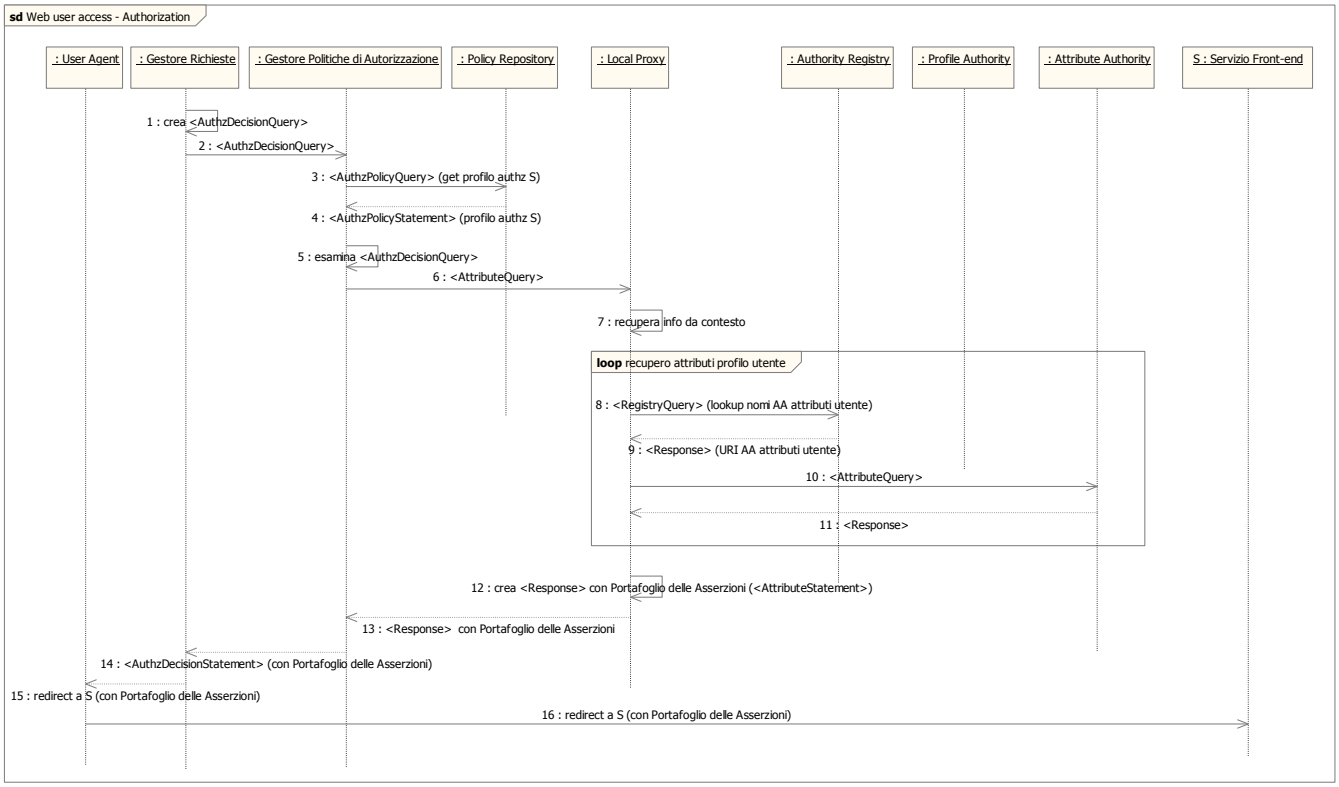

Figura 23. Scenario di autorizzazione nell'accesso utente via web

I passi della fase di autorizzazione, nell'ipotesi che l'utente sia correttamente autenticato, sono i seguenti.

1. Il Gestore delle Richieste crea un'opportuna richiesta di autorizzazione (<AuthzDecisionQuery>) da indirizzare al Gestore delle Politiche di Autorizzazione.

2. Il Gestore delle Richieste inoltra la richiesta di autorizzazione (<AuthzDecisionQuery>) al Gestore delle Politiche di Autorizzazione.
3. Il Gestore delle Politiche di Autorizzazione interroga il Policy Repository per conoscere il profilo di autorizzazione (policy) del servizio S a cui l'utente vuole accedere.
4. Il Policy Repository restituisce al Gestore delle Politiche di Autorizzazione il profilo di autorizzazione del servizio SA.
5. Il Gestore delle Politiche di Autorizzazione esamina la richiesta di autorizzazione ricevuta, sulla base dell'utente che vuole accedere al servizio e della policy di accesso al servizio.
6. Il Gestore delle Politiche di Autorizzazione invia una <AttributeQuery> al Local Proxy (per esempio usando il protocol binding SOAP), indicando in un unico messaggio gli attributi da verificare ai fini dell'autorizzazione dell'utente in base alle politiche di accesso al servizio.
7. Il Local Proxy recupera dal contesto le informazioni relative al profilo dell'utente.
8. Il Local Proxy interroga l'Authority Registry per conoscere l'URI di ciascuna Attribute Authority in grado di certificare gli attributi del profilo utente (<RegistryQuery>).
9. L'Authority Registry risponde restituendo al Local Proxy l'URI delle Attribute Authority richieste (<Response>).
10. Il Local Proxy interroga ciascuna Attribute Authority per ottenere la certificazione degli attributi dall'utente (mediante una <AttributeQuery>).
11. Ciascuna Attribute Authority risponde restituendo una risposta (<Response>) con le asserzioni relative agli attributi dell'utente.
12. Il Local Proxy estrae le asserzioni ricevute e costruisce una risposta (<Response>) globale per il Gestore delle Politiche di Autorizzazione contenente le asserzioni di attributo raccolte (ciò costituisce il Portafoglio delle Asserzioni).
13. Il Local Proxy restituisce al Gestore delle Politiche di Autorizzazione la risposta (<Response>) globale contenente le asserzioni relative a tutti gli attributi dichiarati dall'utente nel proprio profilo.
14. Il Gestore delle Politiche di Autorizzazione fornisce al Gestore delle Richieste la decisione di autorizzazione completa del Portafoglio di Autorizzazione.
15. Il Gestore delle Richieste permette all'utente l'accesso al servizio richiesto, per esempio tramite redirect verso la pagina target.
16. Lo User Agent accede al servizio di front-end richiesto.

6.2.2. *Accesso a servizio in cooperazione applicativa*

Come già illustrato nella sez. 4.4.2, lo scenario in cooperazione applicativa riguarda l'interazione, a seguito di una richiesta da parte di un utente tramite User Agent, tra un servizio applicativo che si trova in un certo dominio (detto dominio richiedente, da cui segue la notazione *_DR) che deve invocare un servizio applicativo posto in un dominio differente (detto dominio erogante, da cui segue la notazione *_DE). L'interazione applicativa che coinvolge il componente del layer INF-1 che si occupa della

comunicazione (trasporto) interdominio non viene mostrata nei diagrammi che seguono ed è trattata a sé stante nella sez. 6.2.2.3.

6.2.2.1. Approccio "push"

Lo scenario che segue mostra l'interazione in cooperazione applicativa secondo un approccio "push" alla trasmissione del Portafoglio delle Asserzioni dell'utente. In altre parole, si suppone qui che il servizio applicativo del dominio richiedente sia in grado raccogliere e quindi di trasmettere al servizio applicativo del dominio erogante un Portafoglio delle Asserzioni completo e sufficiente a permettere l'erogazione del servizio. Ciò può avvenire in due modi:

- la descrizione della policy di autorizzazione al servizio applicativo del dominio richiedente, presente nel Policy Repository del dominio richiedente, include già anche i requisiti di accesso del servizio applicativo del dominio erogante che dovrà essere invocato;
- il Portafoglio delle Asserzioni viene completato in un secondo tempo dal servizio applicativo del dominio richiedente, prima dell'interazione in cooperazione applicativa, accedendo al Policy Repository del dominio erogante e accedendo in particolare alla descrizione della policy di accesso del servizio applicativo del dominio erogante.

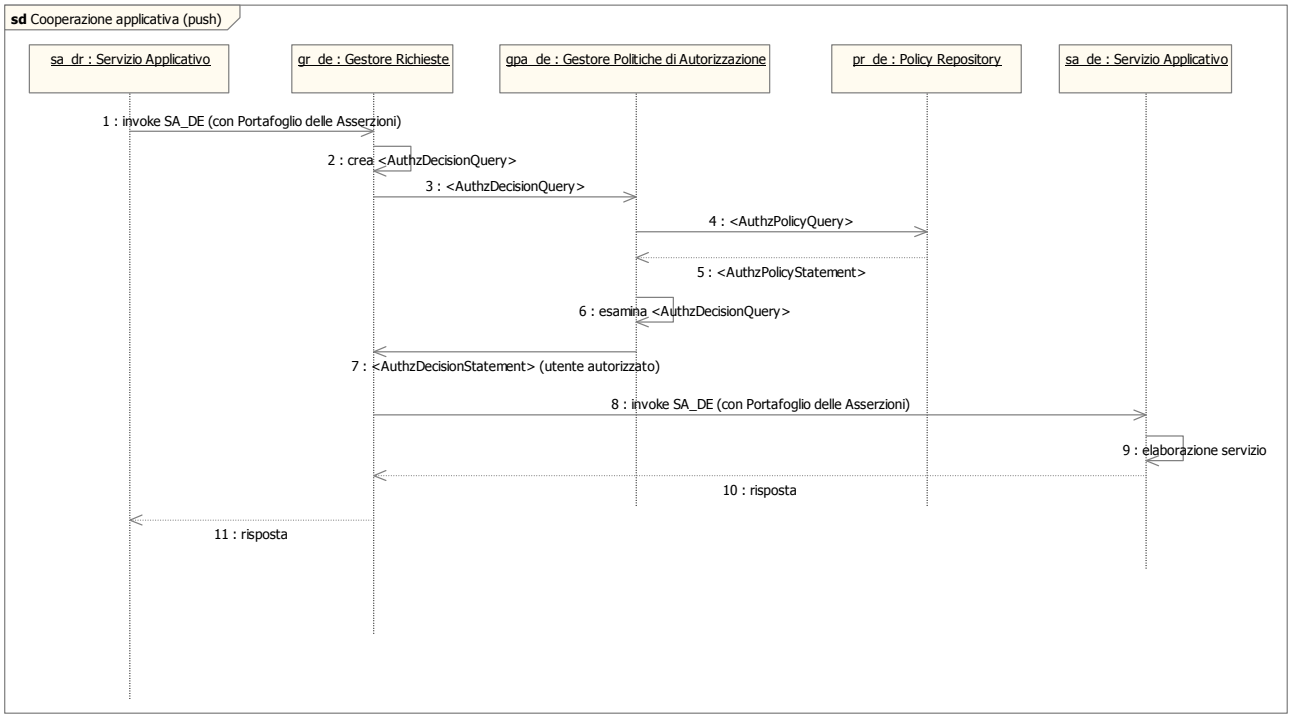

Figura 24. Scenario in cooperazione applicativa, approccio "push"

I passi dell'interazione in cooperazione applicativa secondo l'approccio "push" sono i seguenti.

Si suppone che l'utente si sia correttamente autenticato e sia stato autorizzato ad accedere al servizio applicativo del dominio richiedente, e che il Portafoglio delle Asserzioni sia completo e sufficiente a invocare il servizio remoto.

1. Il Servizio Applicativo del dominio richiedente (SA_DR) invoca un Servizio Applicativo del dominio erogante (SA_DE), contattando il rispettivo Gestore delle Richieste e fornendogli il Portafoglio delle Asserzioni raccolto per l'utente corrente (per maggiori informazioni su questo aspetto si veda la sez. 4.2.8).
2. Il Gestore delle Richieste crea un'opportuna richiesta di autorizzazione destinata al Gestore delle Politiche di Autorizzazione.
3. Il Gestore delle Richieste inoltra al Gestore delle Politiche di Autorizzazione una richiesta di decisione di autorizzazione al servizio corredata dal Portafoglio delle Asserzioni dell'utente.
4. Il Gestore delle Politiche di Autorizzazione del dominio erogante contatta il Policy Repository per conoscere il profilo di accesso del SA_DE.
5. Il Policy Repository fornisce il profilo del SA_DE.
6. Il Gestore delle Politiche di Autorizzazione esamina la richiesta di decisione di autorizzazione.
7. Essendo il Portafoglio delle Asserzioni completo e sufficiente per accedere al SA_DE, il Gestore delle Politiche di Autorizzazione restituisce al Gestore delle Richieste una decisione di autorizzazione, corredata del Portafoglio delle Asserzioni, che permette al SA_DR l'accesso al SA_DE per conto dell'utente.
8. Il Gestore delle Richieste inoltra l'invocazione applicativa al SA_DE, corredata del Portafoglio delle Asserzioni.
9. Il SA_DE elabora la richiesta.
10. Il SA_DE fornisce la risposta applicativa al Gestore delle Richieste.
11. Il Gestore delle Richieste fornisce la risposta applicativa al SA_DR invocante.

6.2.2.2. Approccio "pull"

Come detto, nel caso considerato nella sezione precedente si suppone che il Service Provider del dominio richiedente sia in grado di costruire un Portafoglio delle Asserzioni completo e sufficiente a invocare correttamente il servizio che si trova nel dominio erogante. Ciò è possibile poiché il servizio del dominio richiedente può disporre della descrizione delle politiche di accesso del servizio del dominio erogante e può quindi impostare di conseguenza le fasi di autorizzazione dell'utente tenendo conto anche dei criteri di accesso al servizio remoto (questi passi non sono descritti nella versione attuale del diagramma).

Questo approccio può essere complementato introducendo uno scenario basato su una logica "pull", in cui cioè è il Service Provider del dominio erogante a poter reperire o verificare quando necessario le credenziali in possesso dell'utente che ha innescato l'interazione. Si osserva però che anche in quest'ultimo caso il meccanismo può riguardare solo le asserzioni contenenti statement di attributo e non di identità: queste ultime infatti richiedono necessariamente l'interazione diretta con l'utente,

pertanto devono essere sempre recuperate (ed eventualmente inserite nel Portafoglio delle Asserzioni) prima che possa avere luogo l'interazione in cooperazione applicativa.

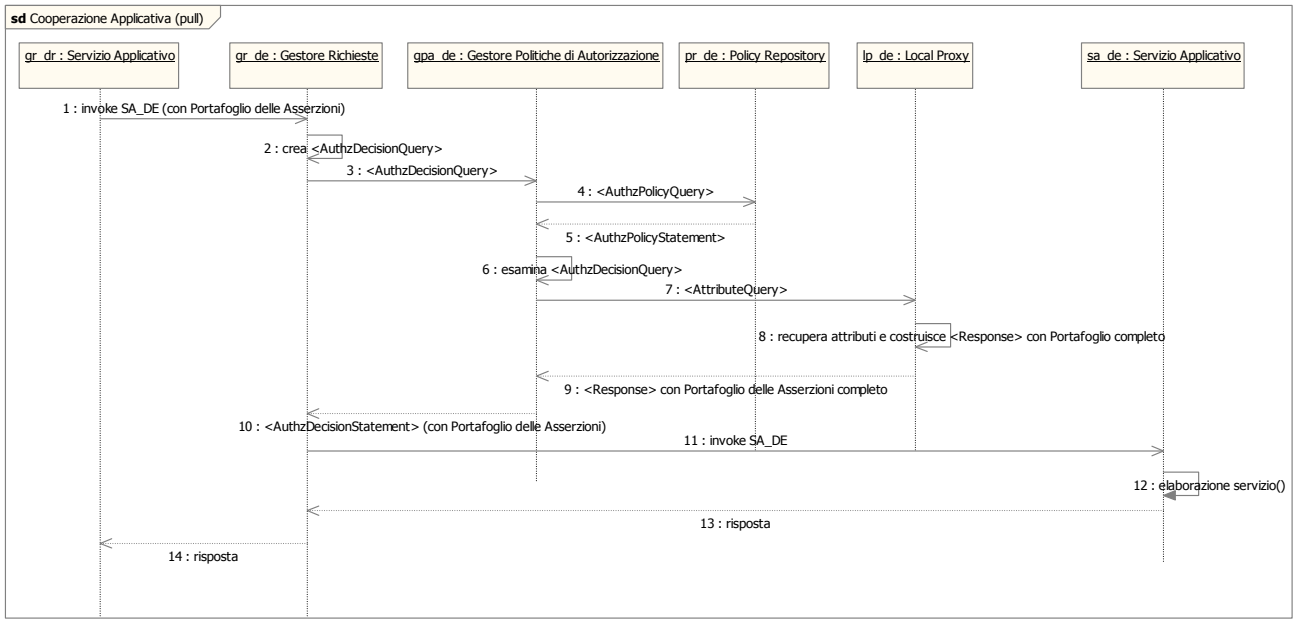

Figura 25. Scenario in cooperazione applicativa, approccio "pull"

I passi dell'interazione in cooperazione applicativa secondo l'approccio "pull" sono i seguenti.

Anche in questo caso, si suppone che l'utente si sia correttamente autenticato e sia stato autorizzato ad accedere al servizio applicativo del dominio richiedente.

1. Il Servizio Applicativo del dominio richiedente (SA_DR) invoca un Servizio Applicativo del dominio erogante (SA_DE), contattando il rispettivo Gestore delle Richieste e fornendogli il Portafoglio delle Asserzioni raccolto per l'utente corrente (per maggiori informazioni su questo aspetto si veda la sez. 4.2.9).
2. Il Gestore delle Richieste crea un'opportuna richiesta di autorizzazione destinata al Gestore delle Politiche di Autorizzazione.
3. Il Gestore delle Richieste inoltra al Gestore delle Politiche di Autorizzazione una richiesta di decisione di autorizzazione al servizio corredata dal Portafoglio delle Asserzioni dell'utente.
4. Il Gestore delle Politiche di Autorizzazione del dominio erogante contatta il Policy Repository per conoscere il profilo di accesso del SA_DE.
5. Il Policy Repository fornisce il profilo del SA_DE.
6. Il Gestore delle Politiche di Autorizzazione esamina la richiesta di decisione di autorizzazione.
7. Poiché il Portafoglio delle Asserzioni fornito non è sufficiente per accedere al SA_DE, il Gestore delle Politiche di Autorizzazione contatta il Local Proxy per reperire e verificare gli attributi utente necessari a completare la fase di autorizzazione dell'utente.

8. Il Local Proxy recupera e verifica gli attributi utente dalle Attribute Authority referenziate nel profilo dell'utente (non si mostrano qui i dettagli di questa fase, si veda a tal proposito lo scenario di autorizzazione descritto nella sez. 6.2.1).
9. Il Local Proxy restituisce al Gestore delle Politiche di Autorizzazione il Portafoglio delle Asserzioni completo.
10. Il Gestore delle Politiche di Autorizzazione restituisce al Gestore delle Richieste una decisione di autorizzazione che permette al SA_DR l'accesso al SA_DE per conto dell'utente, corredata del Portafoglio delle Asserzioni.
11. Il Gestore delle Richieste inoltra l'invocazione applicativa al SA_DE, corredata del Portafoglio delle Asserzioni.
12. Il SA_DE elabora la richiesta.
13. Il SA_DE fornisce la risposta applicativa al Gestore delle Richieste.
14. Il Gestore delle Richieste fornisce la risposta applicativa al SA_DR invocante.

6.2.2.3. Interazioni con il layer INF-1

Negli scenari di interazione in cooperazione applicativa descritti nella sezione precedente sono stati trascurati i dettagli relativi alla trasmissione delle invocazioni applicative inter-dominio. Come già illustrato in precedenza, tale trasmissione avviene mediante il layer INF-1.

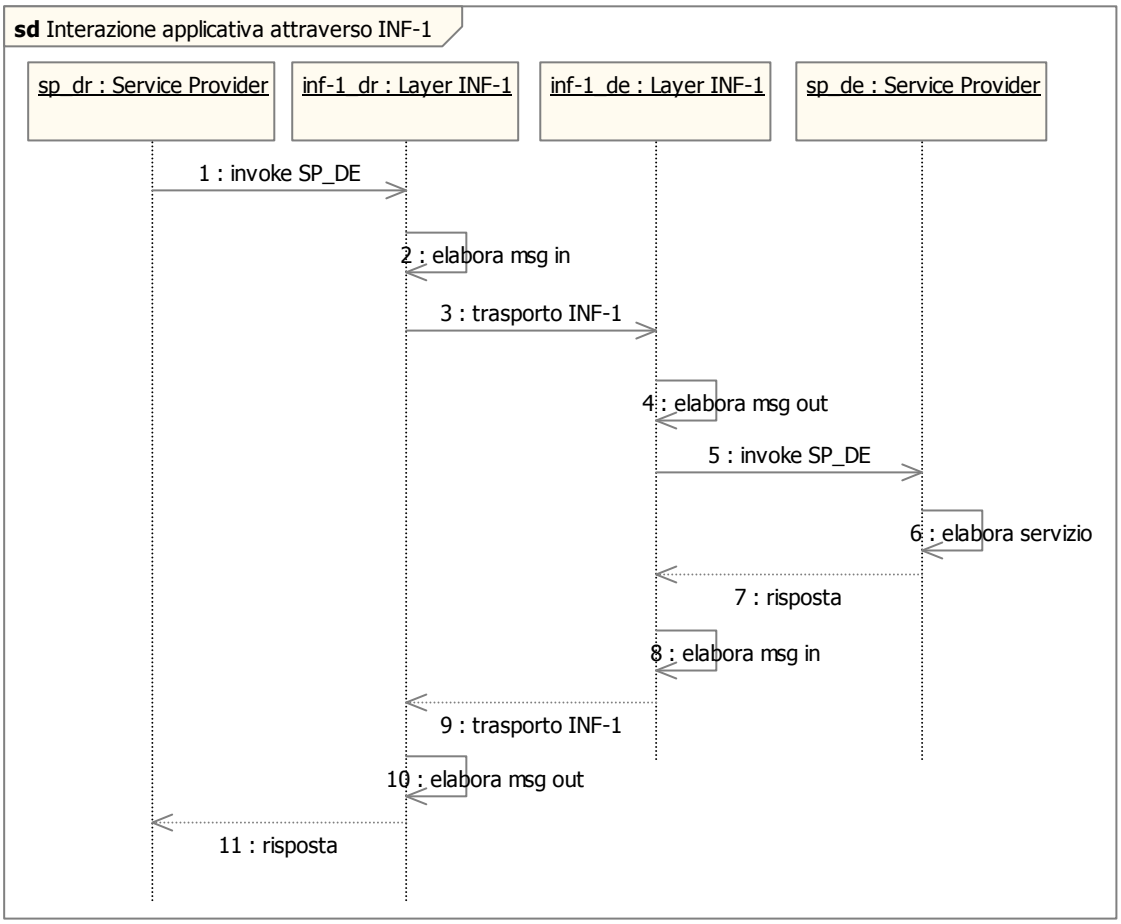

Figura 26. Interazione con il layer INF-1 in caso di cooperazione applicativa

A titolo puramente indicativo, si ripercorrono qui di seguito i passi dell'interazione in cooperazione applicativa che coinvolgono il layer INF-1.

1. Il Service Provider del dominio richiedente comunica al layer INF-1 del suo stesso dominio l'intenzione di invocare un Service Provider di un dominio remoto. A tal fine, il Service Provider fornisce al layer INF-1 l'end-point per poter invocare il servizio remoto e il Portafoglio delle Asserzioni relativo all'utente che ha innescato l'interazione.
2. Il layer INF-1 esamina il messaggio ricevuto dal Service Provider e ne adatta il formato al protocollo di trasporto INF-1 (per esempio, crea una busta di e-government).
3. Il layer INF-1 del dominio richiedente trasmette il messaggio all'omologo layer del dominio erogante.
4. Il layer INF-1 del dominio erogante estrae il contenuto del messaggio ricevuto e lo adatta all'invocazione del Service Provider di destinazione.
5. Il layer INF-1 del dominio erogante inoltra l'invocazione applicativa, opportunamente corredata dal Portafoglio delle Asserzioni contenuto nel messaggio, al Service Provider di destinazione.

6. Il Service Provider del dominio erogante elabora la richiesta applicativa.
7. Il Service Provider del dominio erogante inoltra al layer INF-1 del proprio dominio la risposta applicativa.
8. Analogamente a quanto visto prima, il layer INF-1 adatta la risposta applicativa alla trasmissione attraverso il protocollo di trasporto INF-1.
9. La risposta applicativa viene trasportata mediante INF-1.
10. Giunta al layer INF-1 del dominio richiedente, la risposta applicativa viene adattata all'inoltro al Service Provider invocante.
11. La risposta applicativa giunge infine al Service Provider del dominio richiedente.

6.3. Evoluzione dei profili utente

In aggiunta a quanto detto nella sez. 4.3.6, si può ipotizzare di non porre un limite alla quantità e al tipo di informazioni memorizzate nei profili utente, le quali si trovano sempre sotto il controllo del singolo utente. In questo caso, un'altra considerazione in merito all'uso dei profili utente a fini di autorizzazione riguarda l'uniformità delle informazioni rappresentate: nel caso in cui i profili si trovino sotto il pieno ed esclusivo controllo degli utenti, infatti, si rischia che essi contengano informazioni non omogenee per quel che riguarda il nome, la struttura e il significato degli attributi. Ciò evidentemente complica le fasi di autorizzazione, specialmente in un contesto federato, le quali in estrema sintesi si basano sulla verifica dei valori assunto da certi attributi aventi determinati nomi. Questo problema può essere acuito dallo scenario che prevede che vi possano essere profili multipli archiviati da Profile Authority diverse.

Il problema può essere affrontato in due modi: adottando uno schema/tassonomia/ontologia condiviso (come già segnalato nella sez. 4.3.6 citando le iniziative legate a Shibboleth), oppure prevedendo opportuni meccanismi di "traduzione" da uno schema/tassonomia/ontologia all'altro.

7. BIBLIOGRAFIA

[1] CISIS, Documento di Progetto Interregionale ICAR: "Interoperabilità e Cooperazione Applicativa tra le Regioni", versione definitiva, 7 settembre 2004.

[2] ICAR-INF3, Sistema Federato Interregionale di Autenticazione: Modello concettuale di riferimento, versione 1.0, 2005.

[3] ICAR-INF3, Sistema Federato Interregionale di Autenticazione: Organizzazione, versione 1.0, ottobre 2005.

[4] CNIPA, Sistema pubblico di cooperazione: Quadro Tecnico d'Insieme, versione 1.0, 14 ottobre 2005.
http://www.cnipa.gov.it/site/_files/SPCoop-QuadroInsieme_v1.0_20051014.pdf

[5] CNIPA, Sistema pubblico di cooperazione: Servizi di Sicurezza, versione 1.0, 14 ottobre 2005.
http://www.cnipa.gov.it/site/_files/SPCoop-ServiziSicurezza_v1.0_20051014.pdf

[6] CNIPA, Sistema pubblico di cooperazione: Servizi di Registro, versione 1.0, 14 ottobre 2005.
http://www.cnipa.gov.it/site/_files/SPCoop-ServiziRegistro_v1.0_20051014.pdf

[7] CNIPA, Sistema pubblico di cooperazione: Accordo di Servizio, versione 1.0, 14 ottobre 2005.
http://www.cnipa.gov.it/site/_files/SPCoop-AccordoServizio_v1.0_20051014.pdf

[8] OASIS Security Services (SAML) TC, Security Assertion Markup Language (SAML) V2.0 Technical Overview, Working Draft 08, 12 settembre 2005.
http://www.oasis-open.org/committees/download.php/14361/sstc-saml-tech-overview-2.0-draft-08.pdf

[9] OASIS Security Services (SAML) TC, Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

[10] OASIS Security Services (SAML) TC, Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf

[11] OASIS Security Services (SAML) TC, Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf

[12] OASIS Security Services (SAML) TC, Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf

[13]OASIS Security Services (SAML) TC, Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf

[14]OASIS Security Services (SAML) TC, Security and Privacy Considerations for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 marzo 2005.
http://docs.oasis-open.org/security/saml/v2.0/saml-sec-consider-2.0-os.pdf

[15]OASIS Web Services Security (WSS) TC, Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)
http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

[16]OASIS Web Services Security (WSS) TC, Web Services Security: SAML Token Profile 1.1, OASIS Standard, 1 febbraio 2006.
http://www.oasis-open.org/committees/download.php/16768/wss-v1.1-spec-os-SAMLTokenProfile.pdf

[17]J. Wang, D. Del Vecchio, M. Humphrey, "Extending the Security Assertion Markup Language to Support Delegation for Web Services and Grid Services", Proc. of 2005 IEEE International Conference on Web Services (ICWS 2005), Orlando, FL, July 12-15, 2005.
http://www.cs.virginia.edu/~humphrey/papers/SAML_delegation.pdf

[18]OASIS XACML TC, eXtensible Access Control Markup Language (XACML) Version 2.0, OASIS Standard, 1 febbraio 2005
http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf

[19]OASIS XACML TC, SAML 2.0 profile of XACML v2.0, OASIS Standard, 1 febbraio 2005
http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-saml-profile-spec-os.pdf

[20]Shibboleth Project – Internet2 Middleware, Shibboleth Architecture – Technical Overview, Working Draft 02, 8 giugno 2005
http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-tech-overview-latest.pdf

[21]EDUCAUSE/Internet2 eduPerson task force, eduPerson Specification (200312), dicembre 2003
http://www.nmi-edit.org/eduPerson/internet2-mace-dir-eduperson-200312.html