Guida alla Console di Gestione

Release 3.3.6.p1

Link.it

17 mar 2022

Indice

CAPITOLO 1

Introduzione

Questo manuale documenta le funzionalità e le modalità d'uso della *Console di Gestione* del prodotto *GovWay* (http://gowway.org).

Nota: Oltre alla console di Gestione, GowWay mette a disposizione dei gestori una seconda console utilizzata per il monitoraggio delle richieste applicative gestite dal gateway. Per informazioni sulle modalità di utilizzo della Console di Monitoraggio si rimanda alla relativa manualistica distribuita con il prodotto.

Nel prosieguo si assume che il prodotto GovWay sia già correttamente installato e la console di gestione sia accessibile via browser dai Gestori del Sistema.

L'indirizzo standard della Console di Gestione è *http://ip:porta/govwayConsole*, che dovrà essere correttamente perfezionato con ip e porta del proprio ambiente di installazione. Per informazioni sulle modalità di installazione si rimanda alla relativa manualistica distribuita con il prodotto.

Nota: L'accesso alle diverse funzionalità della console è sempre mediato da un sistema di autorizzazione che verifica che l'utente sia in possesso dei dovuti permessi. Le istruzioni operative sulla gestione degli utenti e la configurazione dei permessi sono descritte nella sezione *Utenti*.

1.1 I Profili di Interoperabilità

GovWay si differenzia dagli API Gateway tradizionali per essere progettato in conformità con i principali profili di interoperabilità in uso nella Pubblica Amministrazione italiana ed europea. Per tale motivo, le modalità di configurazione del prodotto si differenziano in funzione dello specifico profilo a cui le API debbano conformarsi. I profili di interoperabilità supportati dalla distribuzione standard del prodotto sono i seguenti:

- *API Gateway*: è il profilo di interoperabilità di base che consente di supportare qualunque generica API basata su scambio di messaggi SOAP e REST.

- *ModI*: è il profilo che consente di supportare gli scenari di comunicazione basati sul Modello di Interoperabilità rilasciato da AGID, che fornisce i requisiti per l'integrazione tra il sistema informativo complessivo della Pubblica Amministrazione, Cittadini e Imprese.
- *eDelivery*: è il profilo standard adottato a livello europeo nell'ambito del progetto *CEF*, e basato sul protocollo AS4.
- *SPCoop*: il profilo SPCoop è il profilo basato sull'uso della busta eGov e sulla Porta di Dominio, recentemente deprecato da AGID, ma ancora in uso per la quasi totalità dei servizi centrali erogati dalla Pubblica Amministrazione italiana.
- *Fatturazione Elettronica*: questo profilo supporta le modalità di scambio delle fatture elettroniche, nel formato FatturaPA, veicolate tramite il Sistema di Interscambio.

In fase di installazione possono essere scelti i profili di proprio interesse (per default viene proposto il solo profilo di API Gateway).

Durante l'utilizzo della Console di Gestione è preferibile selezionare il profilo di interoperabilità adeguato in base al tipo di configurazioni sui quali si lavora. La selezione del profilo di interoperabilità, tramite il menu presente in testata (Fig. 1.1), comporta la visualizzazione dei soli elementi dell'interfaccia, e relativi dati, attinenti con tale profilo.

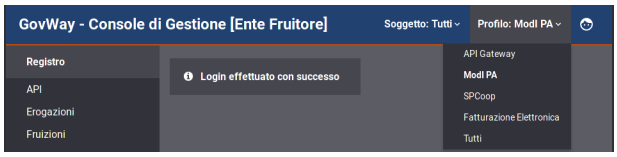


Fig. 1.1: Selezione del profilo di interoperabilità

Esiste la possibilità (non consigliata) di operare sulla console selezionando il profilo *Tutti*. In tal caso non saranno applicati filtri sui contenuti e le maschere di visualizzazione e di configurazione potranno apparire più complesse di quanto avviene selezionando lo specifico profilo su cui si sta lavorando.

Nota: Ulteriori profili sono programmabili in GovWay ed alcuni di questi sono in uso in importanti progetti della pubblica amministrazione, come la Porta di Comunicazione del Sistema di Interscambio del Mercato dell'Energia.

1.2 Le entità di configurazione dei servizi

Prima di descrivere le entità di configurazione presenti nel registro è importante chiarire il concetto di *Dominio* cui alcuni elementi di configurazione fanno riferimento. Il dominio rappresenta il confine logico (tipicamente un ente amministrativo) entro il quale sono racchiuse le risorse applicative da condividere con l'esterno. Nel seguito si fa distinzione tra i seguenti:

- *Dominio Gestito*: l'insieme delle risorse applicative i cui flussi di comunicazione sono sotto il controllo del GovWay di propria gestione.
- *Dominio Esterno*: Insieme di risorse applicative esterne al dominio gestito.

Le principali entità di configurazione del Registro sono:

- *API*

  Descrizione formale dei flussi di comunicazione previsti da un dato servizio, erogato o fruito nel proprio dominio. Ad ogni API è assegnata una singola modalità operativa e, in base ad essa, sarà fornita una descrizione

formale delle interfacce di dialogo supportate. Ad esempio saranno forniti WSDL/XSD per le interfacce Soap o un file YAML in formato Swagger per quelle Rest.

- *Erogazione*

  Registrazione di una specifica istanza di servizio che un soggetto del dominio interno eroga in accordo alle interfacce applicative descritte da un set di API censito nel registro.

- *Fruizione*

  Registrazione di una specifica istanza di servizio che un soggetto del dominio interno fruisce in accordo alle interfacce applicative descritte da un set di API censito nel registro.

- *Soggetto*

  Entità che rappresenta la singola organizzazione, o ente amministrativo, coinvolto nei flussi di comunicazione. Ciascun soggetto censito nel registro può appartenere al dominio interno o esterno e può avere associata un'unica modalità operativa.

- *Applicativo*

  Entità per censire i client, riferiti ad uno specifico soggetto (e quindi modalità), che fruiscono di servizi. Censire un applicativo è indispensabile nei casi in cui l'identificazione è necessaria per poter superare i criteri di autenticazione autorizzazione specificati nella configurazione del *Controllo degli Accessi* per ciascun servizio fruito.

- *Ruolo*

  Entità per censire i ruoli che possono essere utilizzati nell'ambito del controllo degli accessi per costruire specifici criteri di autorizzazione. I ruoli possono avere origine interna al registro oppure essere passati da un sistema esterno, sia in contesti fruizione che di erogazione.

- *Scope*

  Entità per censire gli scope che possono essere utilizzati nell'ambito del controllo degli accessi per costruire specifici criteri di autorizzazione basato sui token.

1.3 Il processo di configurazione dei servizi

Le sezioni successive del documento illustrano i passi necessari per realizzare le configurazioni necessarie per rendere operativi i flussi di erogazione/fruizione dei servizi nei diversi profili di interoperabilità supportati.

Per semplificare il processo di configurazione, nel caso di configurazioni per l'interoperabilità con le note piattaforme di erogazione di servizi centralizzate, GovWay mette a disposizione specifici package, denominati *Govlet*. Il Govlet, attraverso un modello di tipo wizard, consente all'utente di fornire i dati necessari a produrre le entità di configurazione per uno specifico servizio. I Govlet disponibili possono essere acquisiti dal sito di Govway al seguente indirizzo http://www.govway.org/govlets. Alcuni esempi di Govlet:

- *FatturaPA - Fatturazione Attiva*: configurazione del servizio per l'invio di fatture elettroniche al Sistema d'Interscambio (SDI).
- *FatturaPA - Fatturazione Passiva*: configurazione del servizio per la ricezione di fatture elettroniche dal Sistema d'Interscambio (SDI).
- *SIOPE+*: configurazione del servizio per l'invio degli ordinativi di pagamento alla piattaforma SIOPE+ e ricezione delle relative notifiche e giornale di cassa.
- *pagoPA*: configurazione del servizio per l'accesso alla piattaforma dei pagamenti elettronici pagoPA.

Una volta entrati in possesso del Govlet è necessario eseguirlo sulla govwayConsole tramite la funzione *Importa* descritta nella sezione *Importa*.

Per procedere manualmente alla produzione delle configurazioni per i servizi, si utilizzano le funzionalità presenti nella sezione *Registro* della GovWayConsole. Il processo manuale di configurazione può essere schematizzato nei passi seguenti:

1. *Definizione delle API*. Il primo passo prevede la definizione delle API relative ai servizi che si vogliono utilizzare. In questa fase tipicamente si provvede al caricamento del descrittore formale delle interfacce (WSDL, WADL, . . .).
2. *Registrazione dell'erogazione o fruizione*. Il secondo passo, dopo aver registrato l'API del servizio, prevede la creazione di una Erogazione, o di una Fruizione, a seconda del ruolo previsto nell'interazione col servizio.
3. *Configurazione Specifica*. Le interfacce della GovWayConsole sono state progettate in modo che, il completamento dei primi due passi di configurazione, sia sufficiente a disporre di una configurazione funzionante del servizio. Il terzo, e quindi opzionale passo, consiste nella produzione di tutti i dettaglio aggiuntivi di configurazione che sono necessari alla particolare situazione.

   In questo passo si forniscono i dettagli delle funzionalità aggiuntive, che riguardano:

   - *Controllo degli Accessi*: indicazione dei criteri di autenticazione e autorizzazione necessari per l'accesso al servizio.
   - *Validazione*: processo di validazione dei messaggi in transito sul gateway.
   - *Sicurezza Messaggio*: misure di sicurezza al livello del messaggio richieste.
   - *Tracciamento*: personalizzazione delle tracce prodotte nel corso dell'elaborazione delle richieste di servizio.
   - *Registrazione Messaggi*: indicazione dei criteri di salvataggio degli elementi che compongono le richieste di servizio (payload, header, allegati, . . .).

La Fig. 1.2 descrive lo scenario generale in cui opera GovWay.

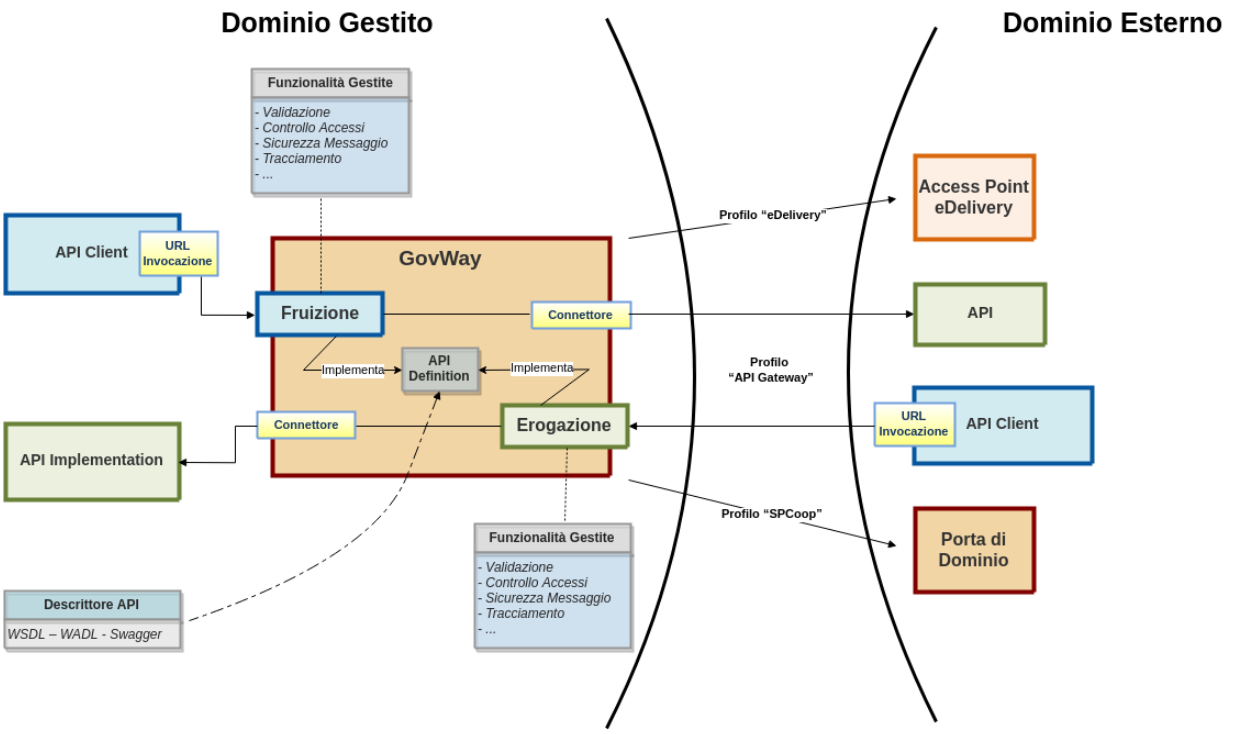


Fig. 1.2: Scenario Generale

La sezioni successive descrivono in dettaglio il processo di configurazione di cui sopra, fornendo i dettagli specifici per ciascun profilo di interoperabilità.

CAPITOLO 2

Profilo “API Gateway”

In questa sezione descriviamo le fasi di configurazione di GovWay al fine di attivare l’erogazione o la fruizione di servizi che rispettano lo standard Soap o Rest. Per semplificare l’utilizzo della console grafica govwayConsole, è consigliabile effettuare la selezione del profilo *API Gateway* tramite l’apposito selettore posto nell’intestazione della pagina.

2.1 Definizione delle API

Indipendentemente che si voglia erogare o fruire un servizio, è necessario iniziare il processo di configurazione con il censimento delle relative API. Questa operazione si effettua sulla govwayConsole posizionandosi nella sezione *Registro* > *API*.

La pagina di ingresso mostra l’elenco delle API eventualmente già presenti in configurazione. Ciascun elemento dell’elenco riporta l’identificativo, il tipo SOAP o REST e il formato del descrittore fornito in configurazione (Fig. 2.1).

Gli elementi dell’elenco possono essere selezionati per l’eliminazione, con il pulsante Elimina, e per l’esportazione, con il pulsante Esporta. La funzione di esportazione è descritta nella sezione *Esporta*.

Si crea una nuova API premendo il pulsante *Aggiungi*.

Compilare il form (Fig. 2.2) inserendo i seguenti dati:

- *Tipo*: Selezionare il tipo delle API a scelta tra «Soap» e «Rest».
- *Nome*: Assegnare un nome che identifichi le API.
- *Descrizione*: un testo opzionale di descrizione.
- *Tags*: un elenco di tag da associare all’API per classificarla. Iniziando a scrivere, vengono proposti i tag già esistenti compatibili.
- *Versione*: progressivo numerico che identifica l’indice di revisione.

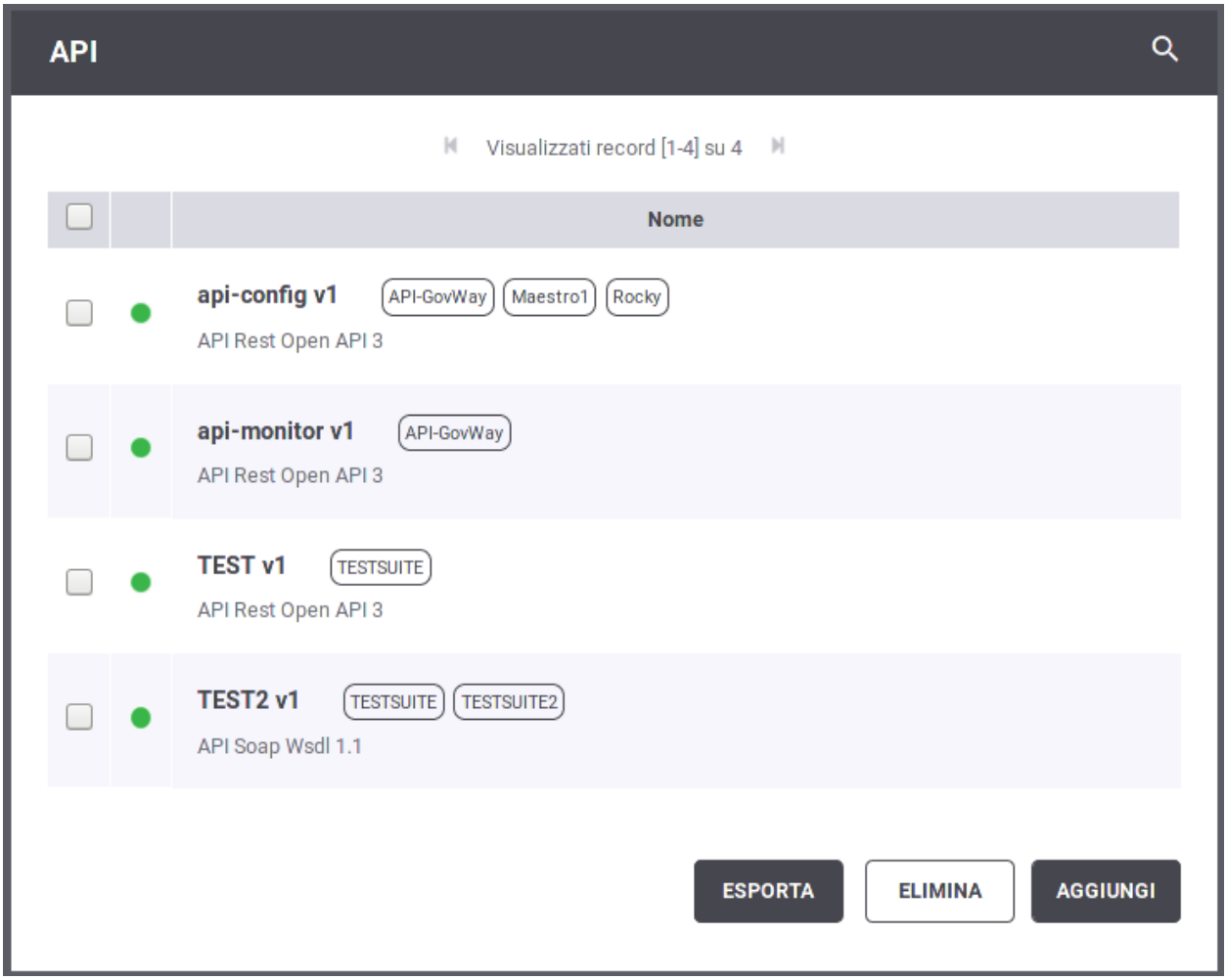


Fig. 2.1: Elenco delle API

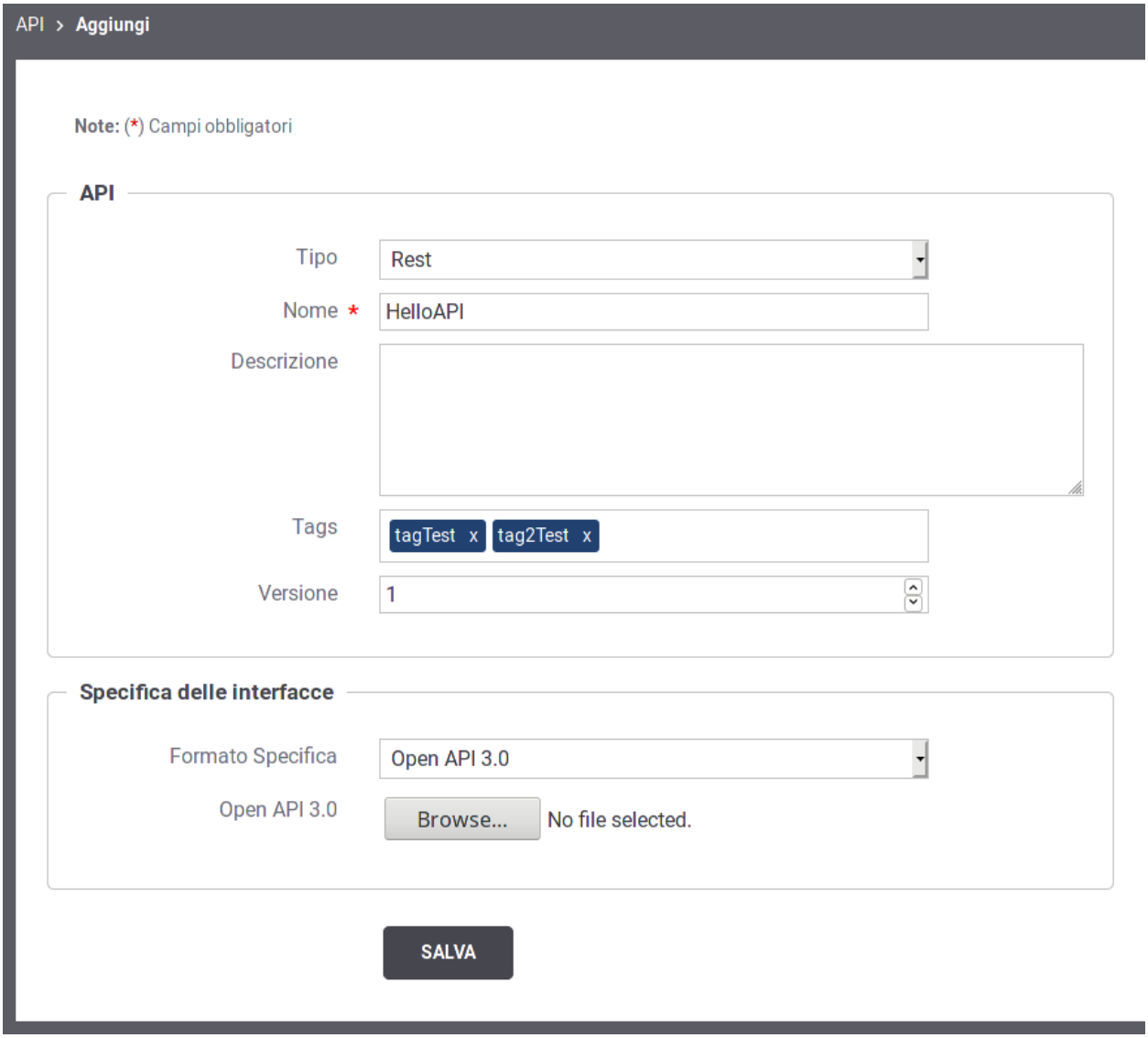


Fig. 2.2: Definizione di una API

- *Specifica delle Interfacce*: In questa sezione è possibile caricare il descrittore formale dell'interfaccia, analizzando il quale, il gateway produce la corrispondente configurazione. Nel caso di interfacce Soap si potrà caricare il relativo WSDL. Nel caso di interfacce Rest si potrà scegliere tra i formati: WADL, Swagger 2.x e OpenAPI 3.3.

  Nel caso non si disponga del descrittore dell'interfaccia è sempre possibile inserire manualmente la relativa configurazione seguendo le modalità descritte alla sezione *Configurazione manuale delle interfacce*.

Effettuato il salvataggio, l'API sarà consultabile all'interno dell'elenco delle API registrate. Accedendo al dettaglio si potranno visionare, a seconda del tipo di API SOAP o REST, rispettivamente i servizi o le risorse che tale API dispone. Nella figura Fig. 2.3 viene riporta l'elenco delle risorse di una API REST.

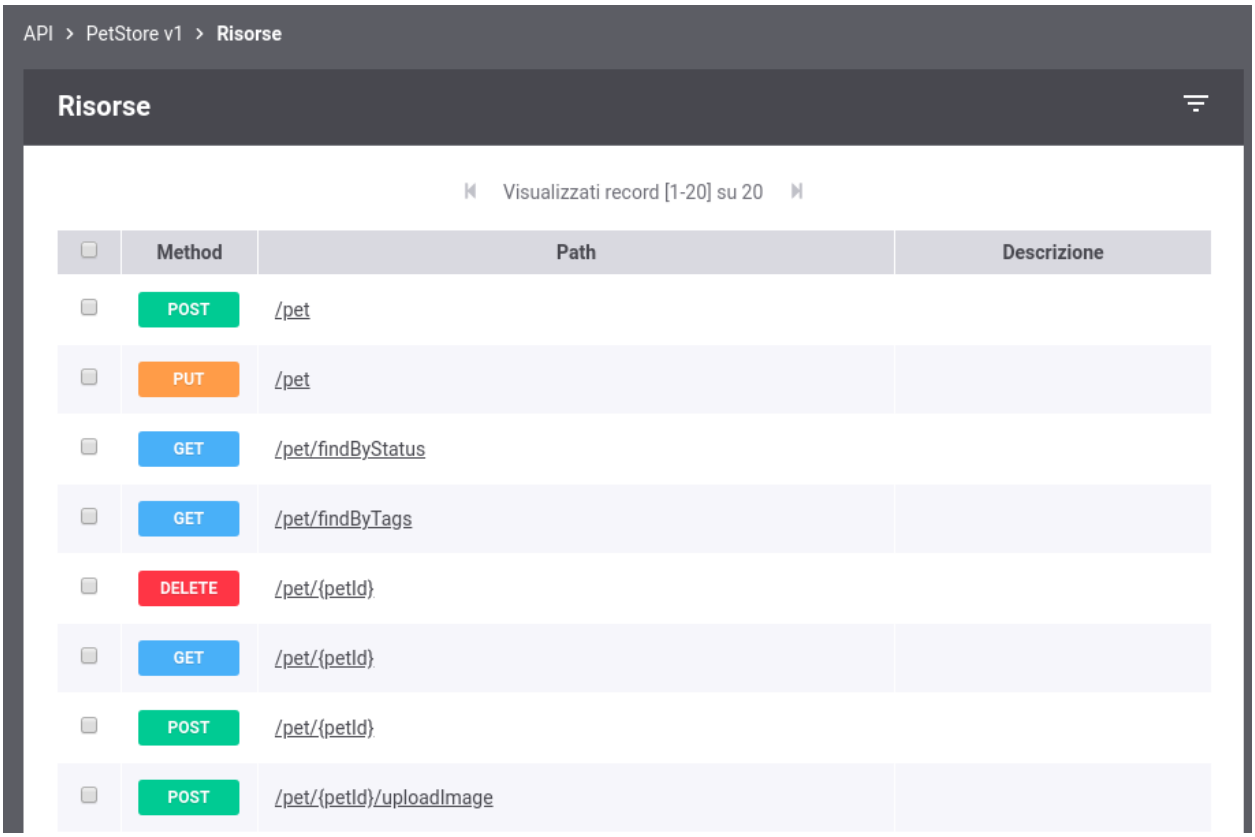


Fig. 2.3: Risorse di una API REST

2.2 Registrazione dell'erogazione

Una volta disponibile la definizione delle API, si passa alla registrazione dell'erogazione fornendo i dati di base per l'esposizione del servizio erogato tramite GovWay. In Fig. 2.4 è illustrato graficamente il caso dell'erogazione.

Per registrare l'erogazione del servizio ci si posiziona nella sezione *Registro* > *Erogazioni* e si preme il pulsante *Aggiungi*.

Compilare il form (Fig. 2.5) inserendo i seguenti dati:

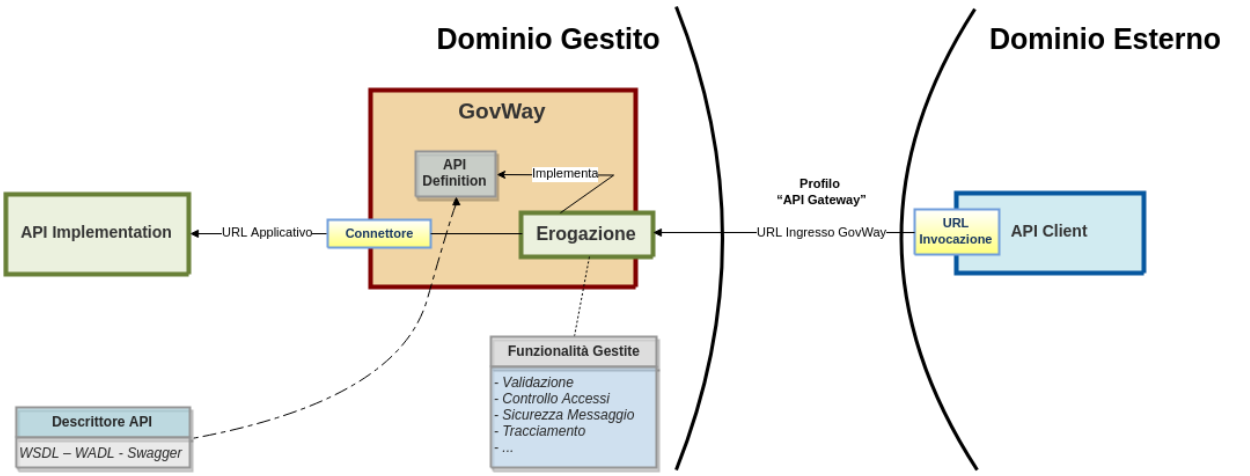


Fig. 2.4: Scenario di riferimento per l'erogazione

- *API - Nome:* Selezionare dall'elenco il nome e la versione relativa alla API cui l'erogazione fa riferimento. Se la API selezionata è di tipo Soap, sarà necessario selezionare anche il Servizio che si vuole erogare.
- *Controllo degli Accessi:* In questa sezione è possibile stabilire l'eventuale controllo degli accessi all'erogazione: - *Pubblico*: non sono richieste credenziali per l'accesso. - *Autenticato*: l'accesso è ammesso solo previa verifica dei criteri di autenticazione e autorizzazione previsti in configurazione.

  Selezionando l'opzione *Autenticato*, dopo la creazione dell'erogazione, sarà necessario completare la configurazione del controllo degli accessi come descritto nella sezione *Autenticazione Trasporto*.
- *Connettore:* In questa sezione devono essere specificati i riferimenti al servizio, al fine di rendere possibile il corretto instradamento delle richieste inviate dai soggetti fruitori. Questo connettore riferisce il servizio del dominio interno che si sta erogando.

  Le informazioni da fornire sono:

  - *Utilizza Applicativo Server:* flag che consente di selezionare un applicativo di tipo «Server» invece di fornire tutte le informazioni relative al connettore. Per i dettagli consultare la sezione *Connettore*.
  - *Endpoint:* la url per la consegna delle richieste al servizio.
  - *Autenticazione Http:* credenziali da fornire nel caso in cui il servizio richieda autenticazione di tipo HTTP-BASIC.
  - *Autenticazione Https:* credenziali da fornire nel caso in cui il servizio richieda autenticazione di tipo HTTPS.
  - *Proxy:* nel caso in cui l'endpoint del servizio sia raggiungibile solo attraverso un proxy, possono essere indicati qui i relativi riferimenti.
  - *Ridefinisci Tempi Risposta:* permette di ridefinire i tempi di risposta che sono stati configurati a livello generale, nell'ambito del controllo del traffico (vedi sezione *Tempi Risposta*)

Nota: Se l'API riferita dall'erogazione possiede un descrittore (WSDL, OpenAPI, ecc.) l'interfaccia propone come valore di default per il connettore l'endpoint del servizio.

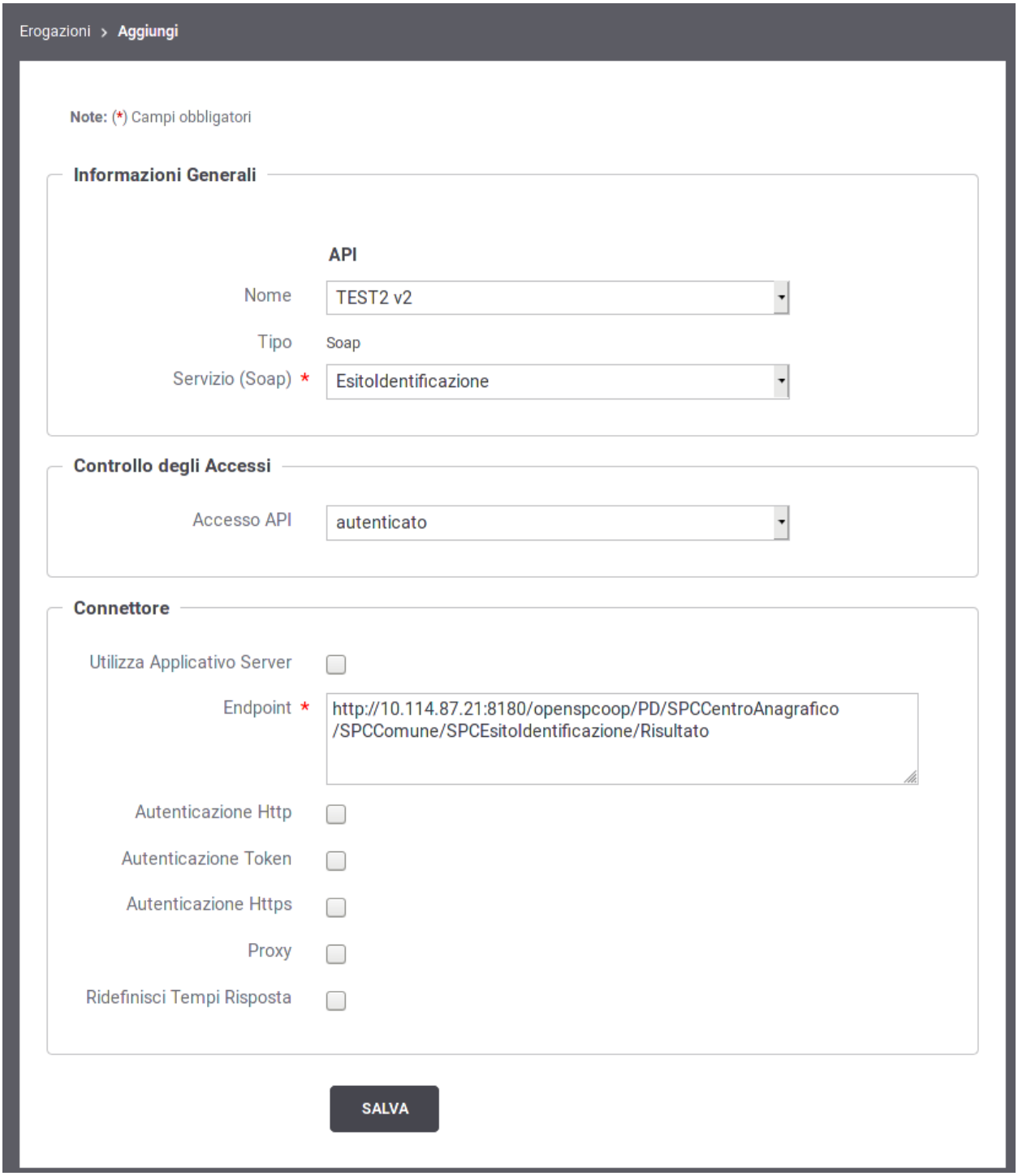


Fig. 2.5: Registrazione di una Erogazione

2.2.1 Completamento configurazione e indirizzamento del servizio

Dopo aver definito le API e registrato la relativa erogazione, come descritto nelle sezioni precedenti, si dispone della configurazione di un servizio erogato i cui riferimenti possono essere comunicati ai fruitori.

Per aggiungere ulteriori dettagli di configurazione, o semplicemente per conoscere il giusto endpoint cui il fruitore deve indirizzare le richieste, si procede dalla pagina di dettaglio dell'erogazione già creata. Il dettaglio dell'erogazione si raggiunge andando alla sezione del menu *Registro* > *Erogazioni*, cliccando sull'elemento visualizzato nell'elenco delle erogazioni presenti nel registro (Fig. 2.6).

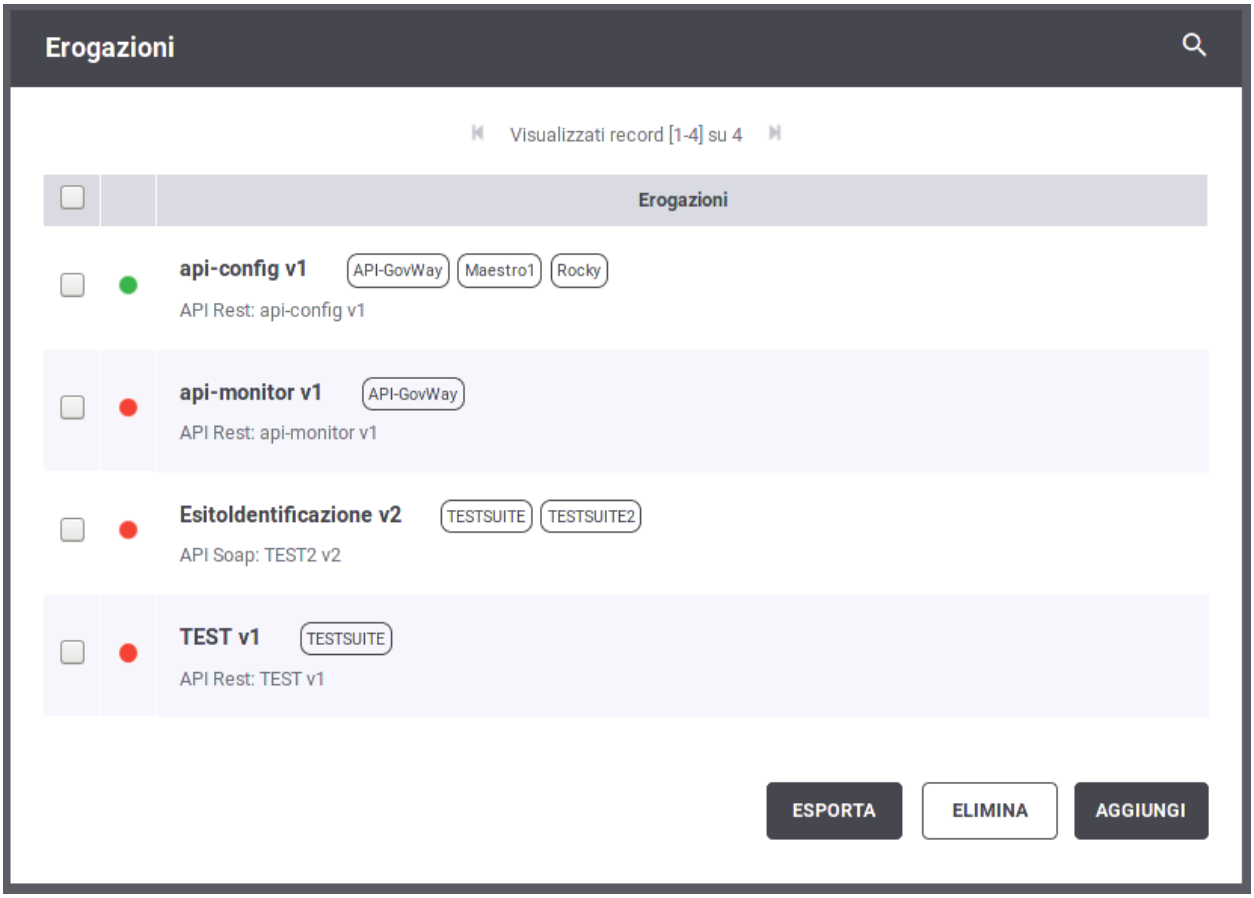


Fig. 2.6: Elenco Erogazioni presenti nel registro

Per la ricerca dell'elemento nell'elenco delle erogazioni è possibile filtrare i dati visualizzati tramite la maschera di filtro che compare cliccando sulla voce *Erogazioni* nell'intestazione dell'elenco (Fig. 2.7).

Il dettaglio dell'erogazione mostra i dati principali e con le icone «matita» è possibile entrare sulle maschere di editing per effettuare delle modifiche. In corrispondenza del connettore è disponibile anche un pulsante che consente di verificare la raggiungibilità dell'indirizzo impostato. In corrispondenza della API riferita, è possibile accedere al relativo dettaglio aprendo un nuovo tab del browser (Fig. 2.8).

La pagina di dettaglio dell'erogazione visualizza i principali elementi di configurazione, che sono:

Fig. 2.7: Filtro delle Erogazioni presenti nel registro

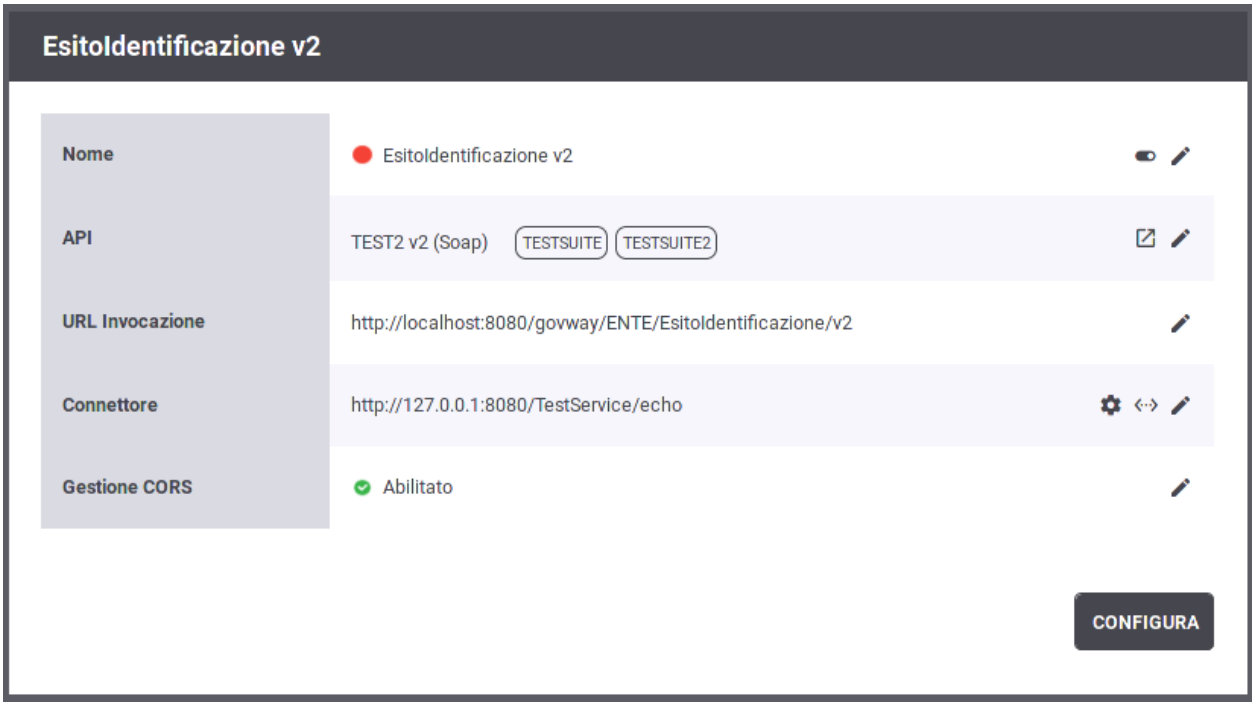


Fig. 2.8: Dettaglio dell'erogazione

- **Nome**: nome dell'erogazione. Accanto al valore è presente l'icona a matita che consente di modificare tale valore. In assenza di configurazioni specifiche per risorsa/azione (sezione *Differenziare le configurazioni specifiche per risorsa/azione*) è presente anche un'icona che permette di disattivare/riattivare l'erogazione. Lo stato di attivazione dell'erogazione è segnalato tramite l'icona colorata presente accanto al nome.
- **API**: API cui fa riferimento l'erogazione con evidenza degli eventuali tags. È presente un'icona che apre in una nuova finestra l'interfaccia per la gestione della configurazione della specifica API.
- **URL Invocazione**: URL che deve utilizzare il mittente per accedere al servizio erogato tramite il gateway. Questo dato rappresenta la *URL* del servizio nel caso Soap o la *Base URL* nel caso Rest. Per la selezione dell'operazione da invocare si distinguono i seguenti casi:
  - *REST*: Indipendentemente che l'API sia stata configurata fornendo il relativo descrittore, WADL o OpenAPI, l'identificazione dell'operation sarà sempre effettuata in automatico dal contesto di invocazione. Non è quindi necessario fornire ulteriori indicazioni.
  - *SOAP*
    - *API con WSDL*: l'operation viene automaticamente identificata dal contesto di invocazione grazie alle informazioni presenti nel descrittore.
    - *API senza WSDL*: l'operation viene identificata inserendo il relativo identificativo nella URL di invocazione, <URL_Invocazione>/<Azione>

    Sono disponibili ulteriori metodi per l'identificazione dell'operation nel caso SOAP, per i cui dettagli si rimanda alla sezione *Modalità di identificazione dell'azione*.
- **Connettore**: Endpoint del servizio acceduto dal gateway, cui verranno consegnate le richieste pervenute. È presente l'icona a matita per aggiornare il valore del connettore. È inoltre presente un'icona che consente di testare la raggiungibilità del servizio tramite il connettore fornito. Maggiori dettagli vengono forniti nella sezione *Connettore*.
- **Gestione CORS**: stato abilitazione della funzione CORS. L'icona a matita consente di modificare l'impostazione corrente come descritto nella sezione *Gestione CORS*.

Ulteriori elementi possono essere indicati per specificare il funzionamento dell'erogazione. Si tratta degli elementi di configurazione specifica, per i cui dettagli si rimanda alla sezione *Configurazione dell'API*.

2.2.2 Condivisione dei dati di integrazione

Le richieste di erogazione, pervenute a GovWay, vengono elaborate e, nel corso dell'operazione, vengono creati i riferimenti alle entità di configurazione presenti nel registro.

GovWay comunica i dati di contesto ricavati, ai sistemi interlocutori, ed in particolare:

- Tutti i dati dell'header di integrazione, relativi al messaggio di richiesta, vengono inviati all'applicativo destinatario (erogatore). I dati che compongono l'header di integrazione sono quelli descritti nelle tabelle presenti alla sezione *Header di Integrazione*.
- Un sottoinsieme dell'header di integrazione, relativo al messaggio di risposta, viene inviato al soggetto mittente (fruitore). I dati inviati (sempre in riferimento alle tabelle della *Header di Integrazione*) sono:
  - *GovWay-Message-ID*
  - *GovWay-Relates-To*
  - *GovWay-Conversation-ID*
  - *GovWay-Transaction-ID*

2.2.3 Errori Generati dal Gateway

La gestione dei casi di errore, nelle comunicazioni mediate da un Gateway, deve tener conto di ulteriori casi di errore che possono presentarsi rispetto al dialogo diretto tra gli applicativi. Oltre agli errori già previsti nelle interfacce dell'API, gli applicativi client possono pertanto ricevere due tipi di errori generati direttamente da GovWay:

- *Errori Client*: identificabili da un codice http 4xx su API REST o da un fault code "Client" su API SOAP. Indicano che GovWay ha rilevato problemi nella richiesta effettuata dal client (es. errore autenticazione, autorizzazione, validazione contenuti...).
- *Errori Server*: identificabili dai codici http 502, 503 e 504 per le API REST o da un fault code "Server" generato dal Gateway e restituito con codice http 500 per le API SOAP.

Per ciascun errore GovWay riporta le seguenti informazioni:

- Un codice http su API REST o un fault code su API SOAP come descritto in precedenza.
- Un codice di errore, indicato nell'header http "GovWay-Transaction-ErrorType", che riporta l'errore rilevato dal gateway (es. AuthenticationRequired, TokenExpired, InvalidRequestContent ...).
- Un identificativo di transazione, indicato nell'header http "GovWay-Transaction-ID", che identifica la transazione in errore, utile principalmente per indagini diagnostiche.
- Un payload http, contenente maggiori dettagli sull'errore, opportunamente codificato per API REST (*REST Problem Details - RFC 7807*) o SOAP (*SOAP Fault*).

Maggiori dettagli, sulla gestione degli errori, sono disponibili nella sezione *Errori generati da GovWay*.

2.3 Registrazione della fruizione

Nel processo di fruizione sono coinvolti i client (o applicativi) interni al dominio che richiedono, tramite accesso sul gateway, un servizio erogato da un soggetto di un dominio esterno.

In Fig. 2.9 è illustrato graficamente il caso della fruizione.

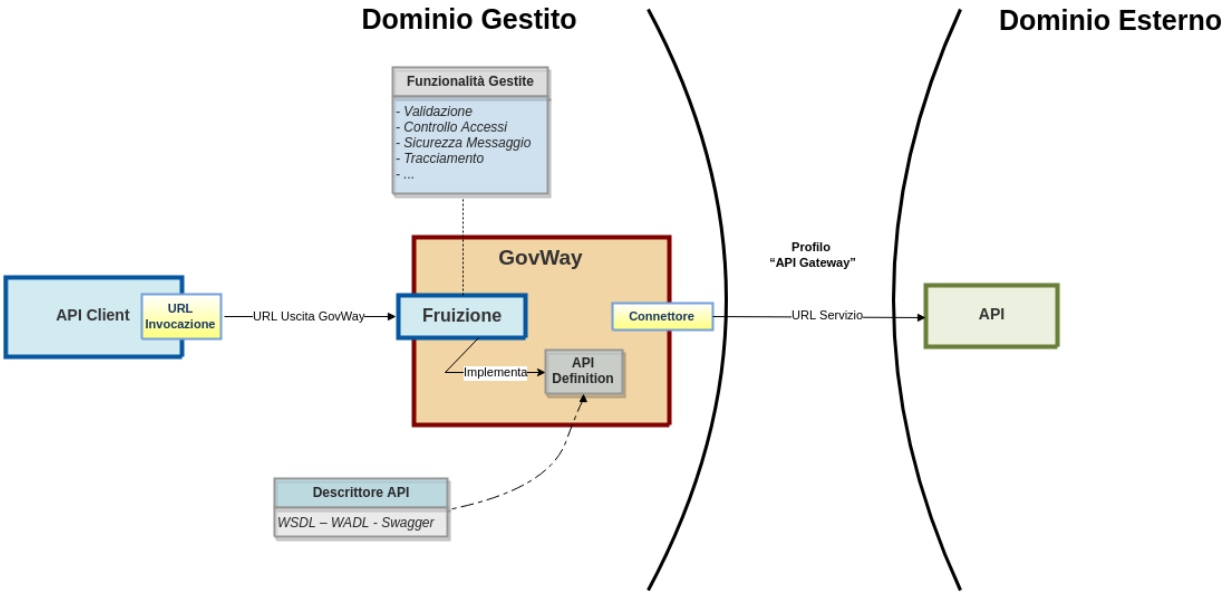


Fig. 2.9: Scenario di riferimento per la fruizione

Analogamente a quanto descritto per le erogazioni, è possibile procedere con la configurazione delle fruizioni accedendo alla sezione di menu *Registro* > *Fruizioni*.

La configurazione delle fruizioni presenta maschere della GovWayConsole del tutto analoghe al caso dell'erogazione. È quindi possibile seguire il processo di configurazione attuando i medesimi passi, illustrati per le erogazioni, calandole sul contesto delle fruizioni.

L'unica differenza, rispetto al processo di configurazione delle erogazioni, è rappresentata dalla presenza del campo *Soggetto Erogatore*, da selezionare come soggetto che eroga il servizio (Fig. 2.10).

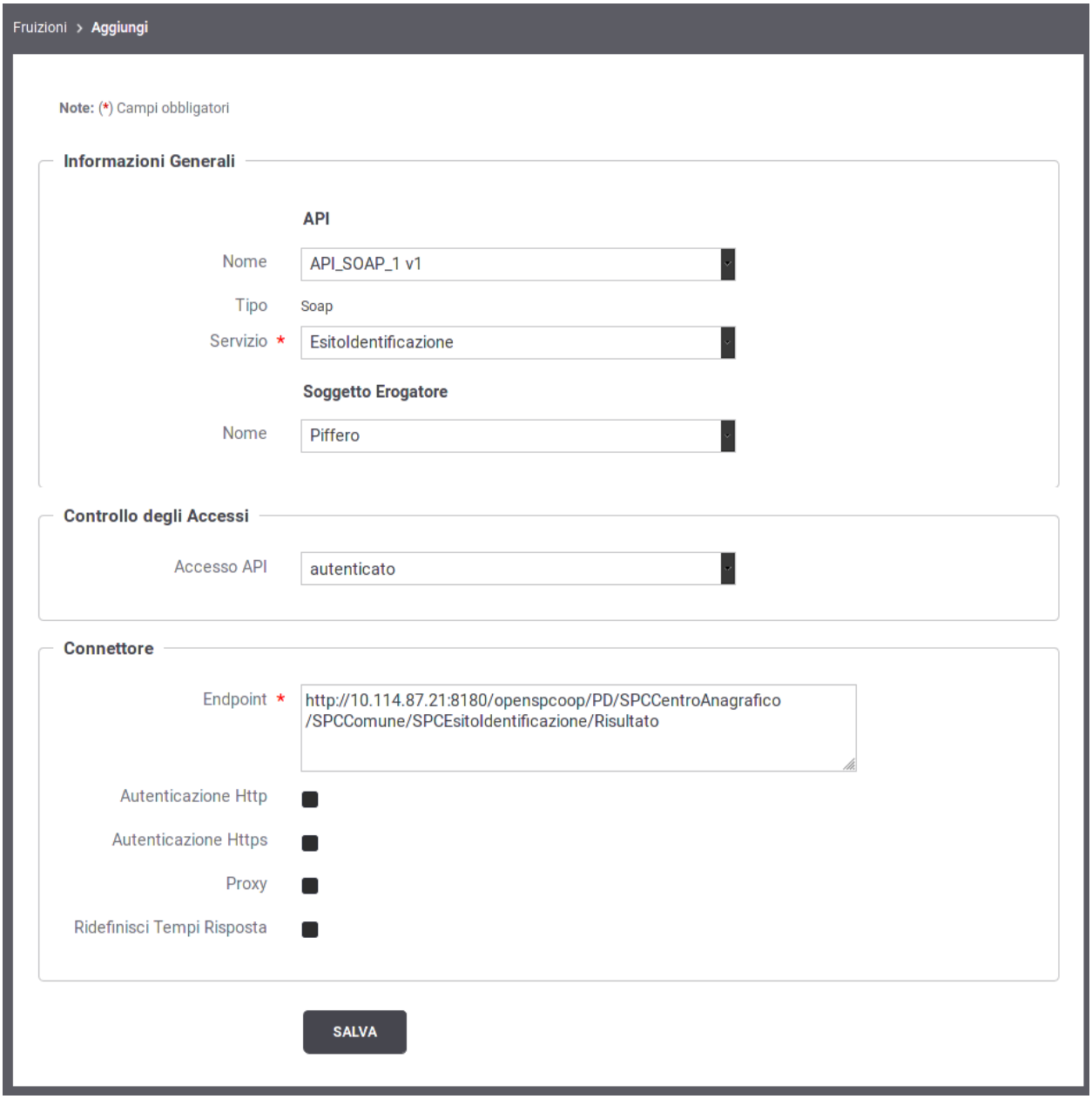


Fig. 2.10: Registrazione di una Fruizione

Nota: Benché non vi siano differenze nelle modalità di configurazione del *Connettore*, nel caso della fruizione questi consiste nei dati di puntamento al servizio erogato sul dominio esterno.

2.3.1 Condivisione dei dati di integrazione

Le richieste di fruizione, pervenute a GovWay, vengono elaborate e, nel corso dell'operazione, vengono creati i riferimenti alle entità di configurazione presenti nel registro.

GovWay comunica i dati di contesto ricavati, ai sistemi interlocutori:

- *GovWay-Message-ID*
- *GovWay-Relates-To*
- *GovWay-Conversation-ID*
- *GovWay-Transaction-ID*

Per ulteriori dettagli si consiglia di consultare la sezione *Header di Integrazione*.

2.3.2 Errori Generati dal Gateway

Analogamente a quanto descritto per le erogazioni, la gestione dei casi di errore nelle comunicazioni mediate da un Gateway devono tenere conto di ulteriori situazioni che possono presentarsi rispetto alla situazione di dialogo diretto tra gli applicativi.

La gestione degli errori viene descritta approfonditamente nella sezione *Errori generati da GovWay*.

2.4 Versionamento delle API

Come descritto nelle precedenti sezioni, ogni API possiede una versione. È possibile registrare una nuova versione dell'API cliccando sul pulsante "Nuova Versione" presente nel dettaglio di una API (Fig. 2.11).

La maschera di creazione della nuova versione non permette ne di modificare il nome dell'API ne di scendere di versione. Vengono ereditati dall'API precedente le altre caratteristiche quali il tipo di API tra SOAP e REST, i tags, la descrizione, il soggetto referente etc.

Se l'opzione "Ridefinisci Interfaccia" è abilitata, viene richiesta una nuova specifica dell'interfaccia dell'API. Terminando la creazione della nuova API verranno creati automaticamente i servizi e le azioni su SOAP o le risorse su REST definiti nella nuova interfaccia (Fig. 2.12).

In alternativa, se l'opzione "Ridefinisci Interfaccia" viene disabilitata, non viene richiesta una nuova specifica di interfaccia e la nuova versione dell'API conterrà la medesima specifica della precendente versione con i medesimi servizi e azioni su SOAP o risorse su REST (Fig. 2.13).

Una volta creata una nuova versione dell'API, è possibile effettuare l'upgrade verso la nuova versione direttamente nell'erogazione e/o nella fruizione che implementa l'API. Infatti se esiste più di una versione di una medesima API è possibile modificarne la versione implementata nell'erogazione o nella fruizione tramite il bottone "modifica" evidenziato nella figura Fig. 2.14.

Accedendo alla modifica è possibile scegliere la versione implementata dell'API, tra le versioni disponibili, come mostrato nella figura Fig. 2.15.

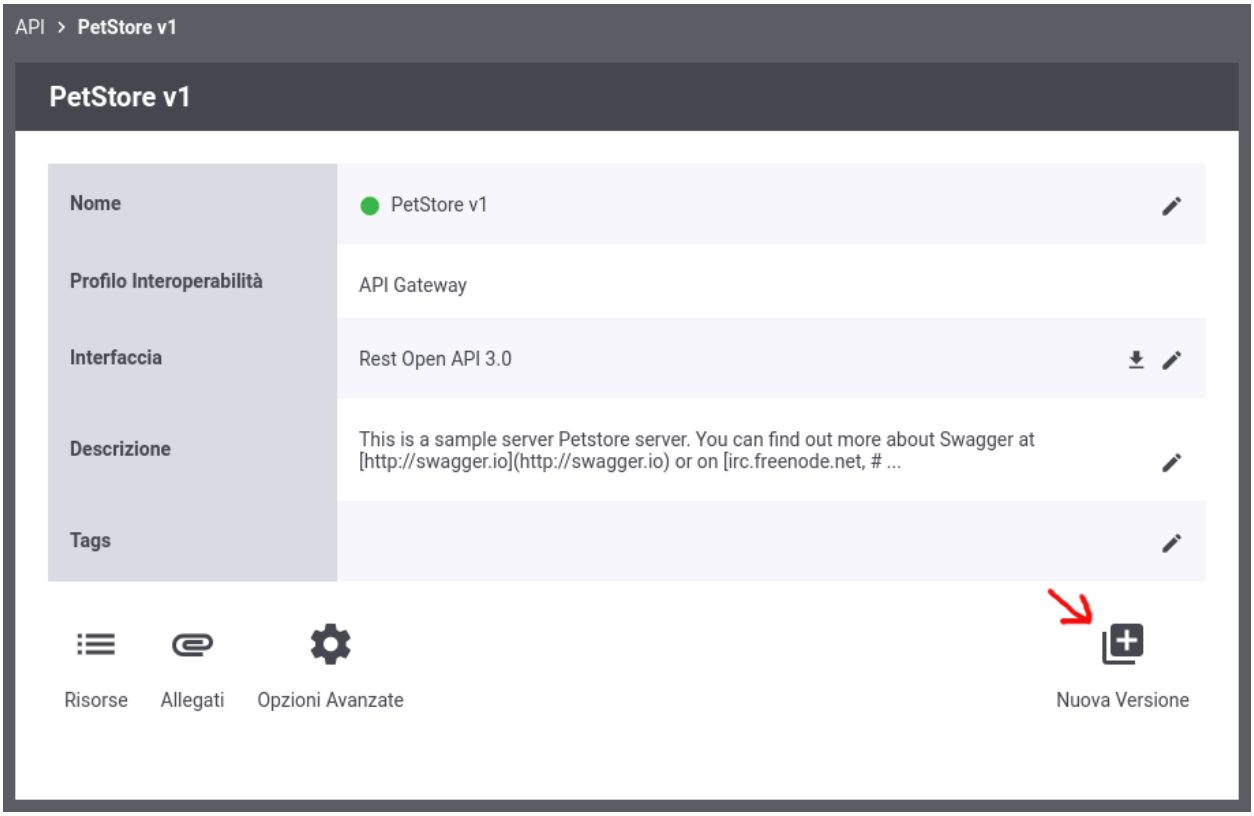


Fig. 2.11: Nuova Versione di una API

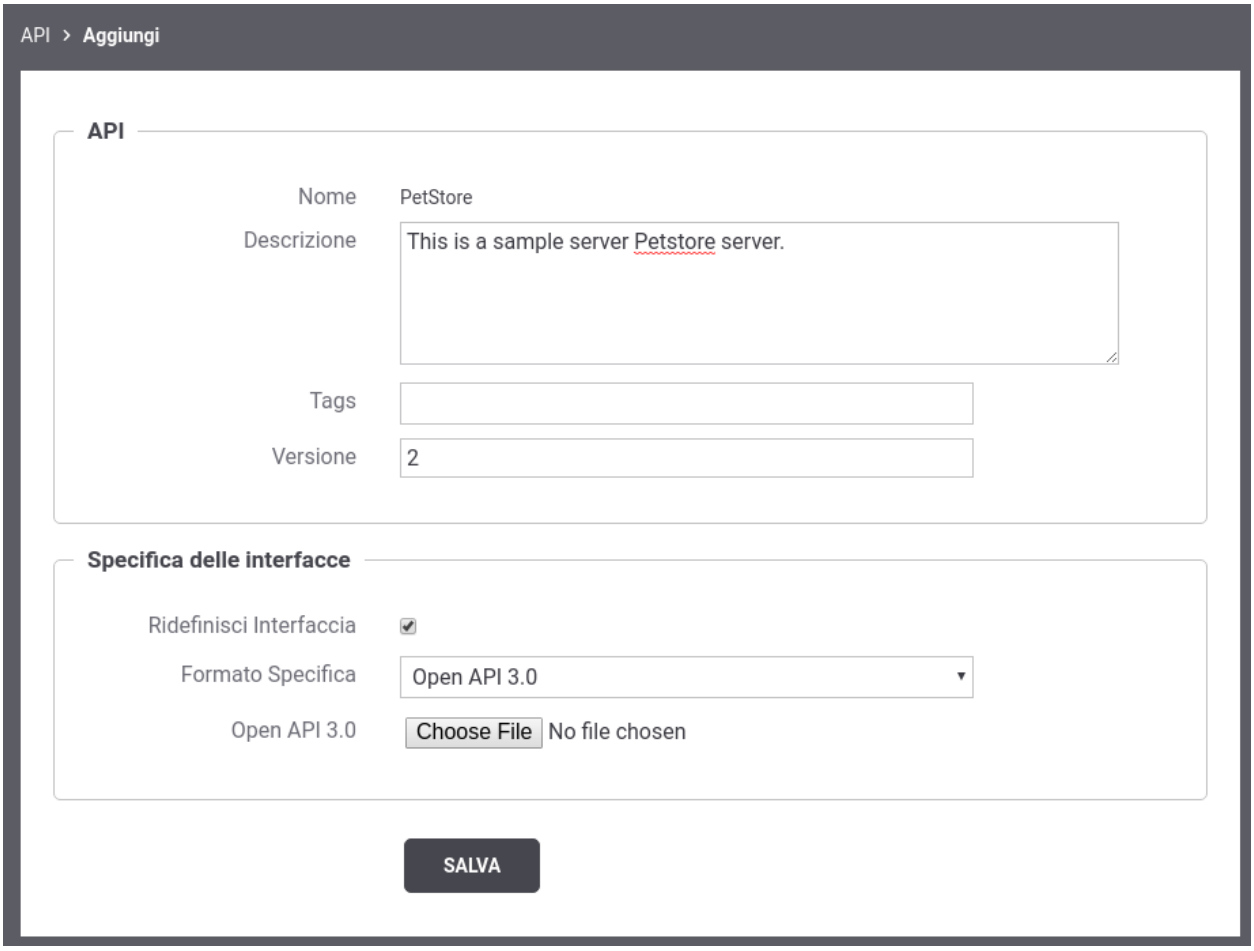


Fig. 2.12: Nuova Versione di una API tramite ridefinizione dell'intervaccia

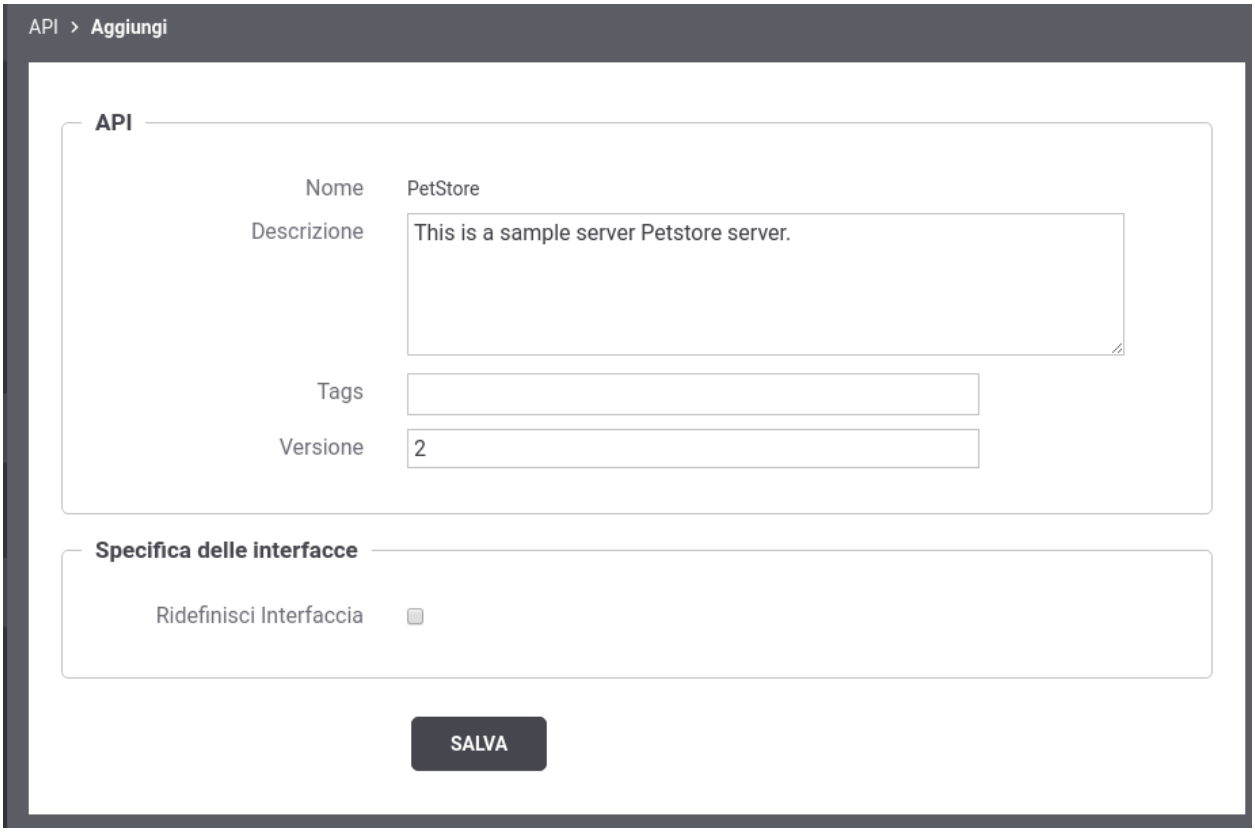


Fig. 2.13: Nuova API che eredita la specifica di interfaccia dalla versione precedente

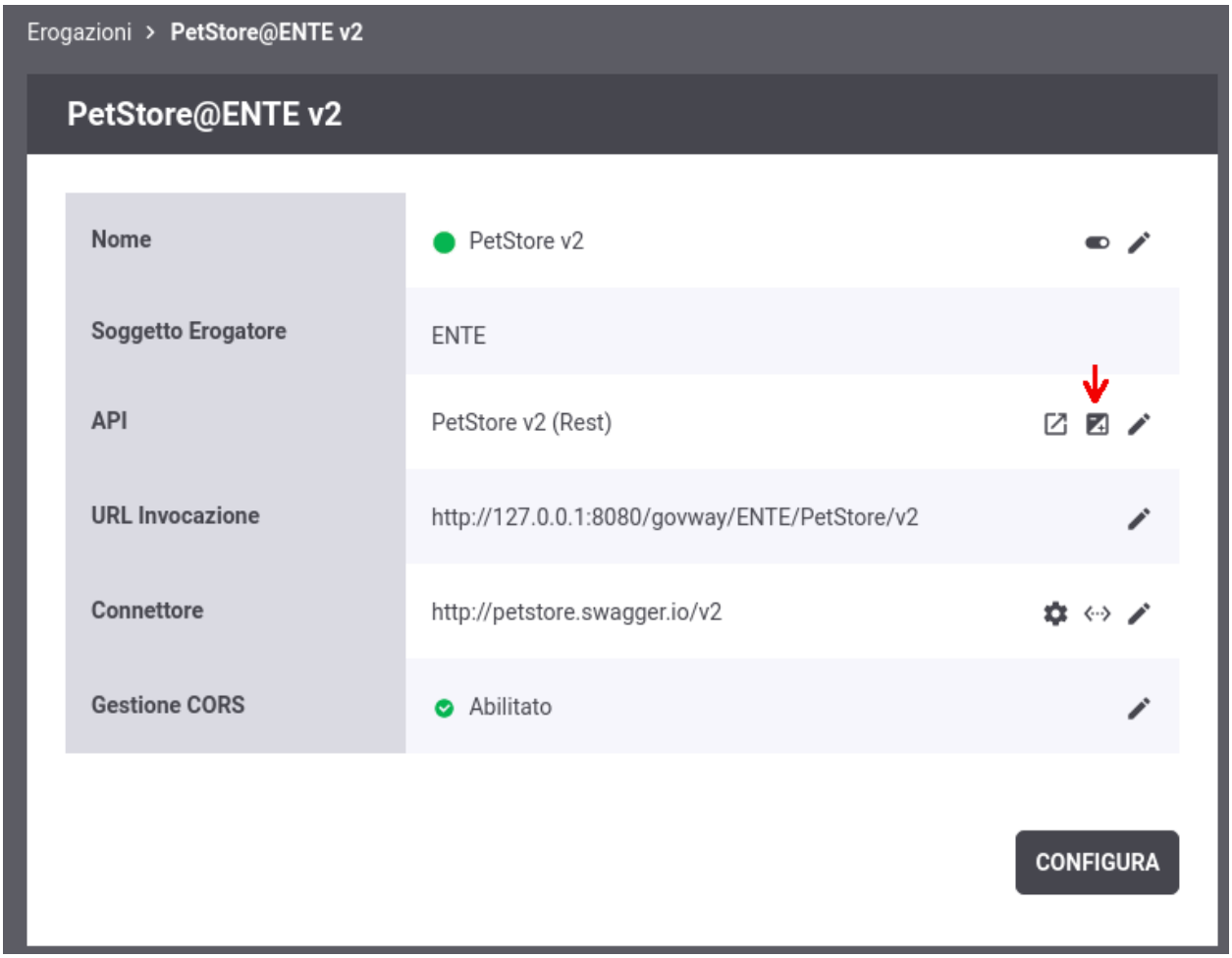


Fig. 2.14: Upgrade di versione dell'API implementata in una erogazione

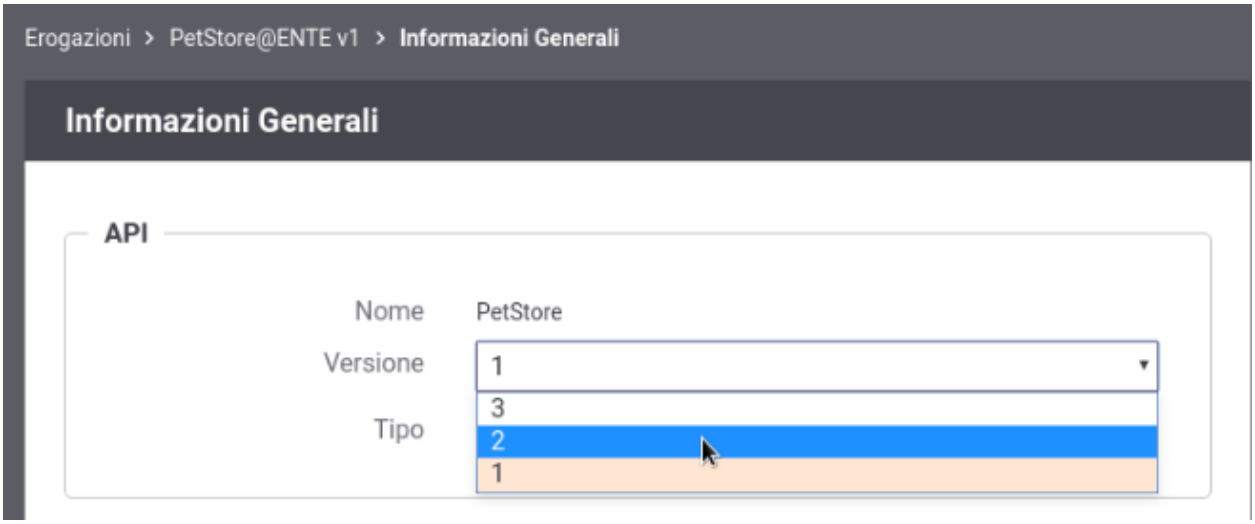


Fig. 2.15: Scelta della versione dell'API implementata in una erogazione

La modifica della versione dell'API implementata dall'erogazione, comporta automaticamente anche la modifica della versione dell'erogazione stessa. Questo si riflette nell'url di invocazione che viene automaticamente aggiornata rispetto alla nuova versione come evidenziato nella figura Fig. 2.16.

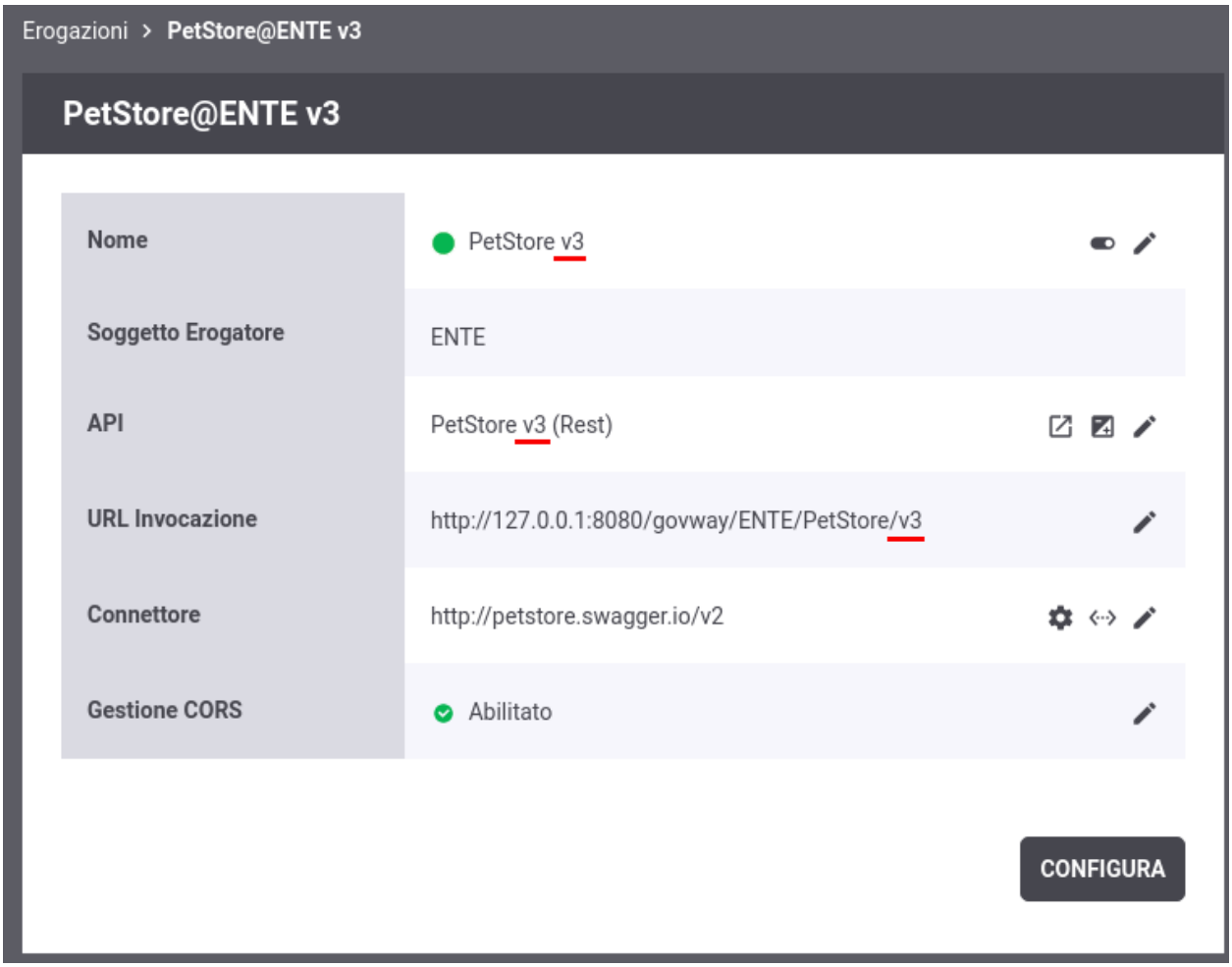


Fig. 2.16: Modifica della versione si riflette sia sull'erogazione che sulla url di invocazione

Nota: Se viene scelta una versione dell'API per la quale esiste già una medesima versione dell'erogazione, il cambio di versione dell'API non si rifletterà sulla versione dell'erogazione e sulla url di invocazione ma solamente sui messaggi scambiati e sulle azioni (soap) o risorse (rest) che l'erogazione espone.

Per maggiori dettagli sul versionamento differente tra erogazione/fruizione ed API e di conseguenza su come questo si riflette nella url di invocazione si rimanda alla sezione *Versionamento delle API e delle Erogazioni/Fruizioni*

2.5 Configurazione dell'API

I passi di configurazione fin qui descritti, per la registrazione di erogazioni e fruizioni, consentono di ottenere uno stato delle entità del registro pronto all'utilizzo in numerose situazioni.

Cliccando sulla voce *Erogazioni* o *Fruizioni* nell'intestazione dell'elenco è possibile consultarne i dettagli selezionando l'API attivata di interesse.

La pagina di dettaglio consente di accedere ai principali elementi di configurazione (Fig. 2.17):

- *Nome*: in assenza di configurazioni specifiche per risorsa/azione (sezione *Differenziare le configurazioni specifiche per risorsa/azione*), accanto al nome dell'erogazione o della fruizione è presente un'icona che permette di disattivare/riattivare l'API come descritto nella sezione *Sospensione API*.
- *URL Invocazione*: se la console viene utilizzata in modalità avanzata (sezione *Modalità Avanzata*), accedendo alla modifica della URL di Invocazione è possibile configurare la modalità di identificazione dell'azione come descritto nella sezione *Modalità di identificazione dell'azione*.
- *Connettore*: endpoint del servizio acceduto dal gateway, cui verranno consegnate le richieste pervenute. In questa è presente sia l'icona a matita per aggiornare il valore del connettore che un'icona che consente di testare la raggiungibilità del servizio tramite il connettore fornito. Maggiori dettagli vengono forniti nella sezione *Connettore*.
- *Gestione CORS*: stato abilitazione della funzione CORS. L'icona a matita consente di modificare l'impostazione corrente come descritto nella sezione *Gestione CORS*.

Tramite il pulsante *Configura* è inoltre possibile aggiungere ulteriori elementi di configurazione attraverso le ulteriori funzionalità messe a disposizione da GovWay (Fig. 2.18).

Le voci di configurazione che possono essere accedute sono:

- *Controllo degli Accessi*
- *Rate Limiting*
- *Validazione dei messaggi*
- *Caching Risposta*
- *Sicurezza a livello del messaggio*
- *MTOM*
- *Trasformazioni*
- *Tracciamento*
- *Registrazione Messaggi*
- *Proprietà*
- *Opzioni Avanzate per Erogazioni/Fruizioni*

Accanto a ciascuna delle voci in elenco è presente un'icona che in base al colore assume i seguenti significati:

- **Grigio**: funzionalità non attiva
- **Rosso**: funzionalità attivata ma configurata in maniera incompleta o errata, quindi non funzionante
- **Giallo**: funzionalità attivata in modalità opzionale o «non bloccante» e quindi in sola notifica
- **Verde**: funzionalità attiva

Le funzionalità specifiche possono essere configurate in maniera differenziata per gruppi di risorse/azioni relative alla API erogata/fruita. Una nuova configurazione specifica può essere creata tramite il pulsante *Crea Nuova*. Il passaggio

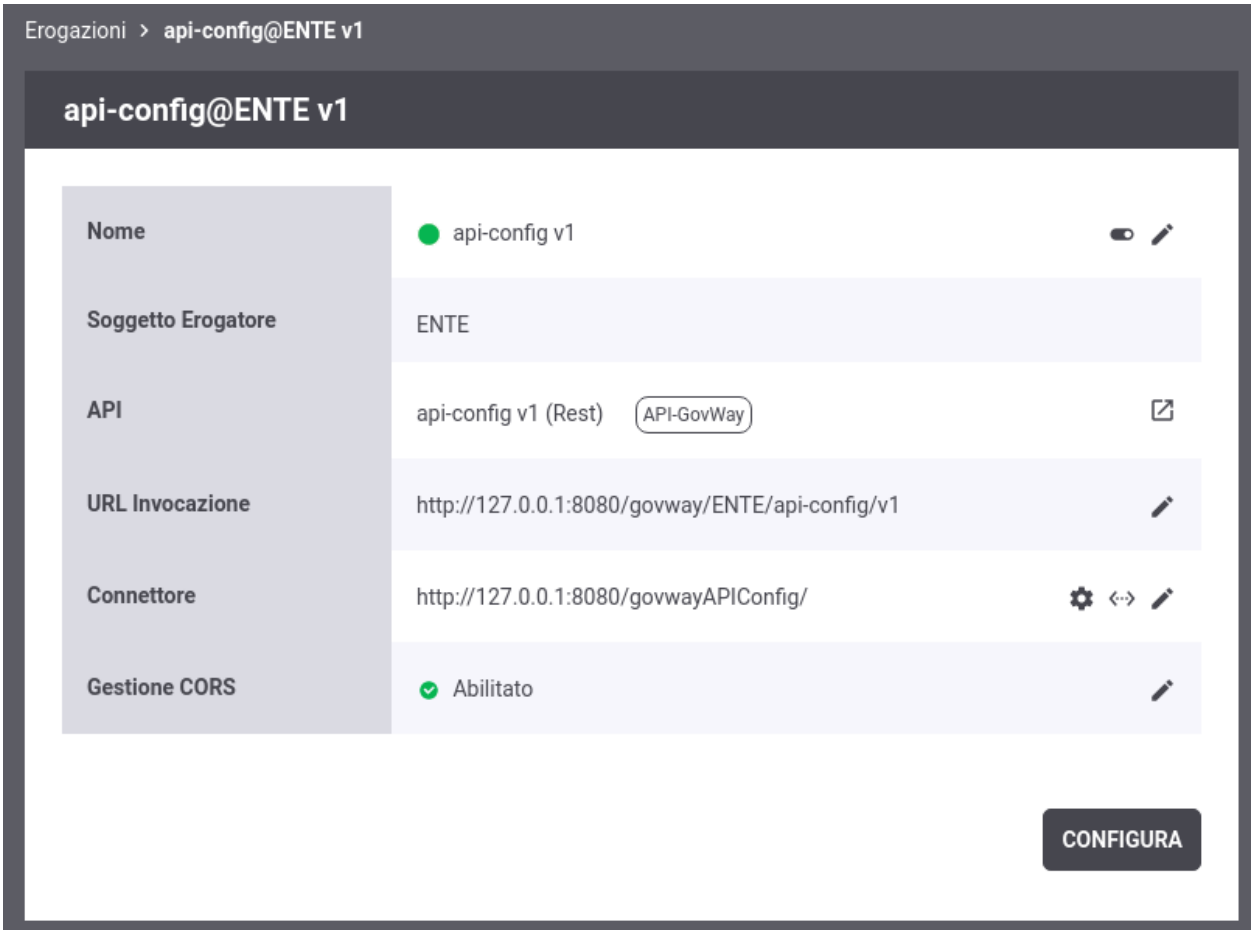


Fig. 2.17: Dettaglio di una erogazione

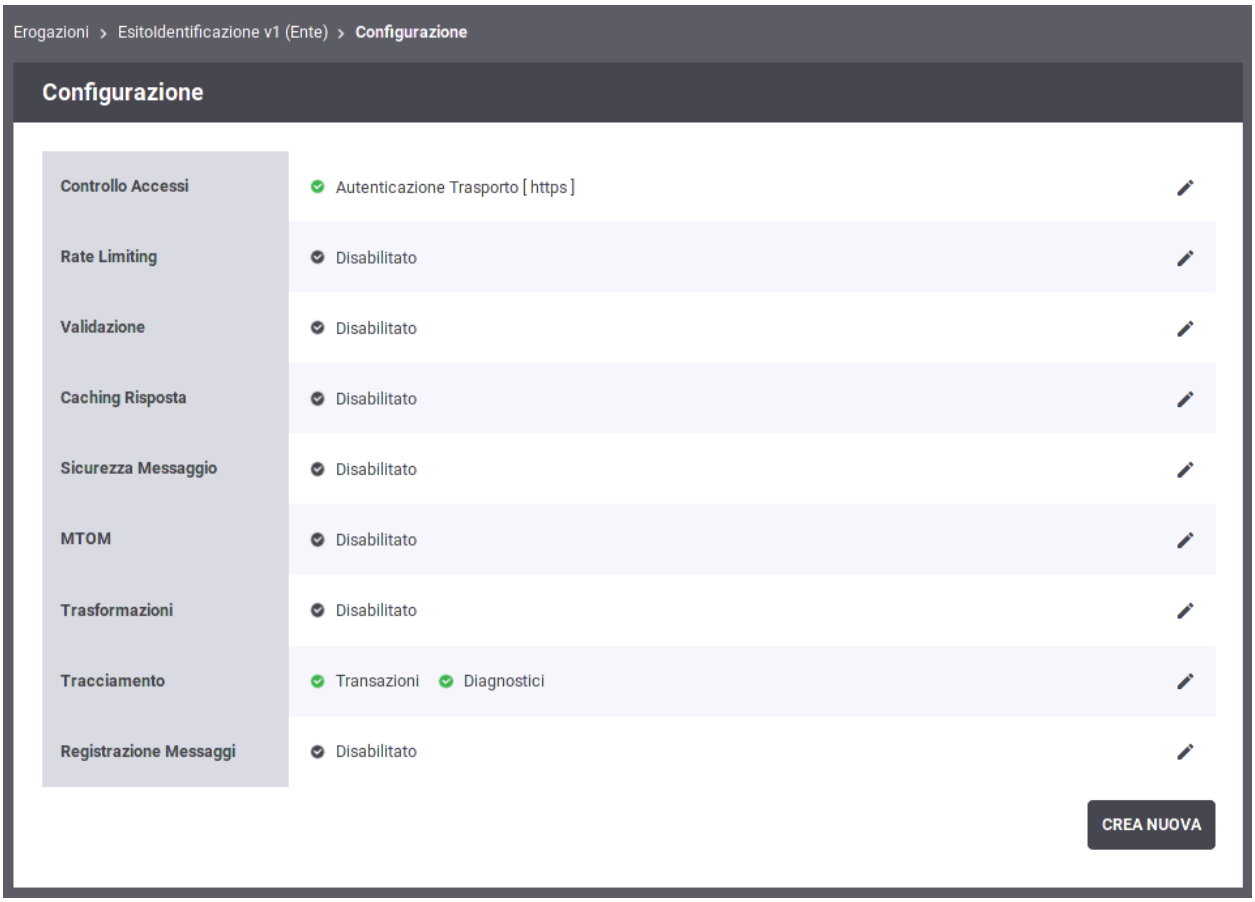


Fig. 2.18: Configurazione di una erogazione

tra una configurazione e l'altra sarà possibile tramite i tab che risulteranno visibili nell'interfaccia. Questa funzionalità è descritta in dettaglio nella sezione *Differenziare le configurazioni specifiche per risorsa/azione*.

Le sezioni successive descrivono in dettaglio le configurazioni sopraelencate e i relativi contesti di utilizzo. Tranne dove esplicitamente dichiarato, gli schemi di configurazione descritti in seguito possono essere attuati sia sulle erogazioni che sulle fruizioni.

2.6 Sospensione API

La console consente di disabilitare temporaneamente una API attiva sul gateway. Successive invocazioni destinate all'API verranno rifiutate generando un codice di errore *APISuspended*.

È possibile sospendere una API cliccando sull'icona toggle presente nella riga *Nome* del dettaglio di una erogazione o fruizione (Fig. 2.19).

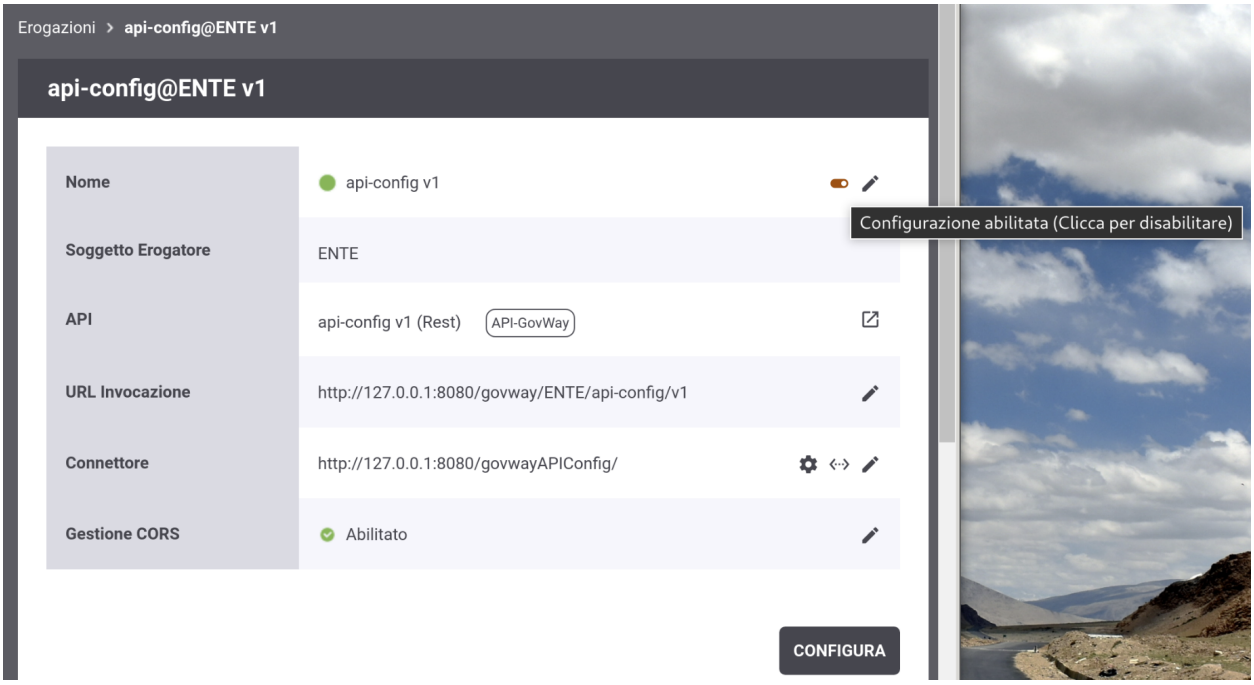


Fig. 2.19: Sospensione di una API

Per poter procedere con la sospensione dell'API l'utente deve confermare l'operazione richiesta come evidenziato nella figura Fig. 2.20.

Quando una API viene sospesa, il suo nome viene affiancato da uno stato rosso che ne evidenzia l'inutilizzo temporaneo. Per abilitarla nuovamente si deve procedere con gli stessi passi effettuati per sospenderla (Fig. 2.21).

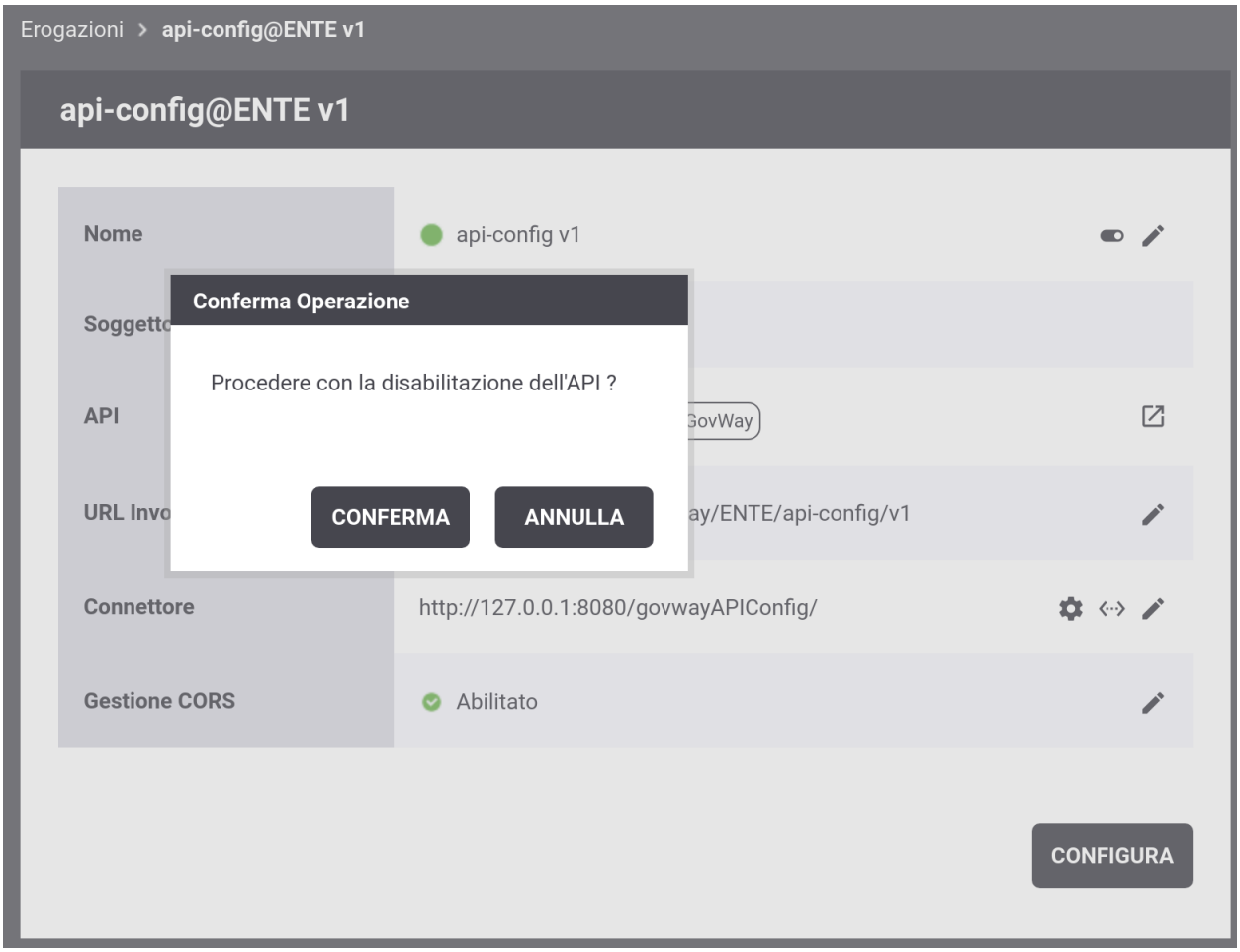


Fig. 2.20: Conferma dell'operazione di sospensione di una API

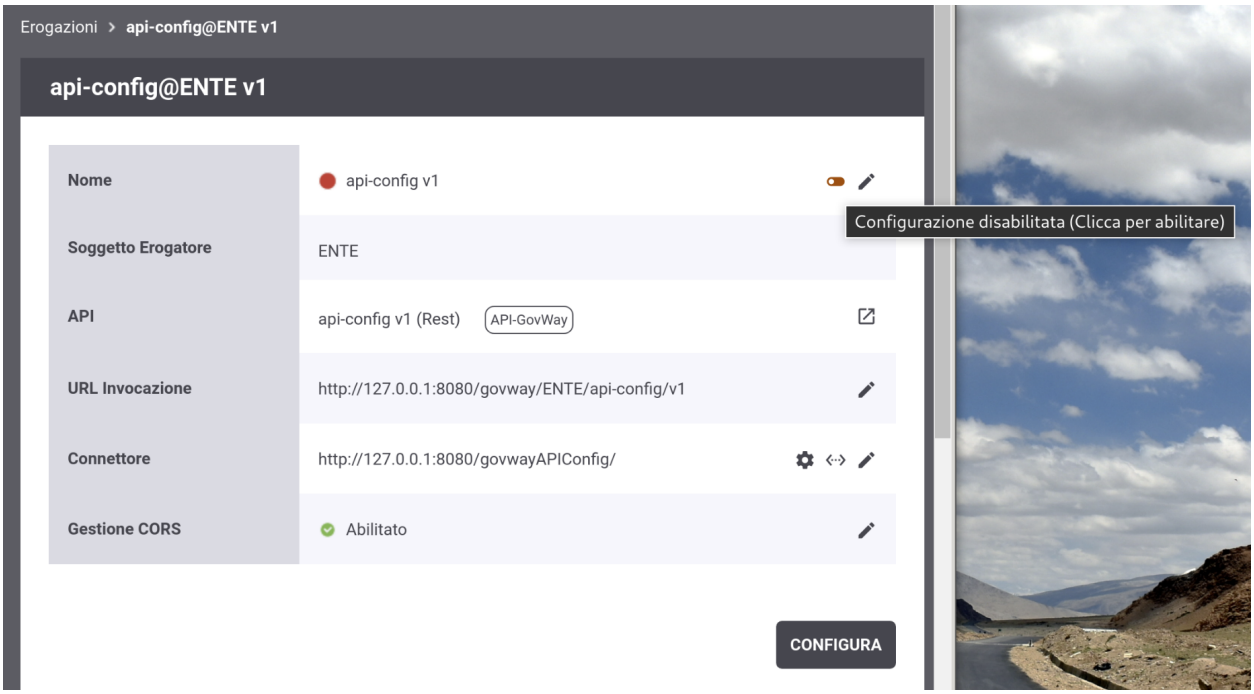


Fig. 2.21: Attivazione di una API

2.7 Connettore

È possibile modificare le impostazioni del connettore (ad esempio per modificare l'endpoint o aggiungere il proxy) seguendo il collegamento presente nella riga *Connettore* del dettaglio di una erogazione o fruizione. I campi del form sono uguali a quelli già descritti per la fase di creazione dell'erogazione (sezione *Registrazione dell'erogazione*). Ulteriori dettagli di configurazione e tipi di connettore diversi da HTTP e HTTPS sono descritti nella sezione *Connettori*.

La sezione *Verifica Connettività Connettore* descrive uno strumento per verificare la raggiungibilità dell'indirizzo impostato.

La sezione *Applicativi Server* descrive invece come censire un'applicazione di backend in modo da poterla riferire su diversi connettori relativi ad erogazioni di API.

Le sezioni successive, infine, descrivono le funzionalità inerenti l'utilizzo di endpoint multipli allo scopo di bilanciare il carico o differenziarlo rispetto a variabili presenti nella richiesta, sempre relativamente ad erogazioni di API.

Nota: Le funzionalità relative ad un applicativo "Server" (sezione *Applicativi Server*) e ai connettori multipli (*Load Balancer* e *Consegna Condizionale*) sono applicabili solamente per le erogazioni di API.

2.7.1 Verifica Connettività Connettore

I contesti in cui l'interfaccia visualizza il valore di un connettore comprendono anche uno strumento per verificare la raggiungibilità dell'indirizzo impostato (Fig. 2.22).

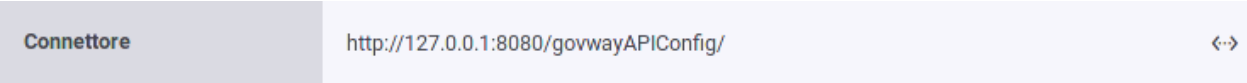


Fig. 2.22: Pulsante per la verifica del connettore

Dopo aver premuto il pulsante si accede ad una schermata che riepiloga le proprietà del connettore e comprende il pulsante *Verifica* per procedere con la verifica (Fig. 2.23).

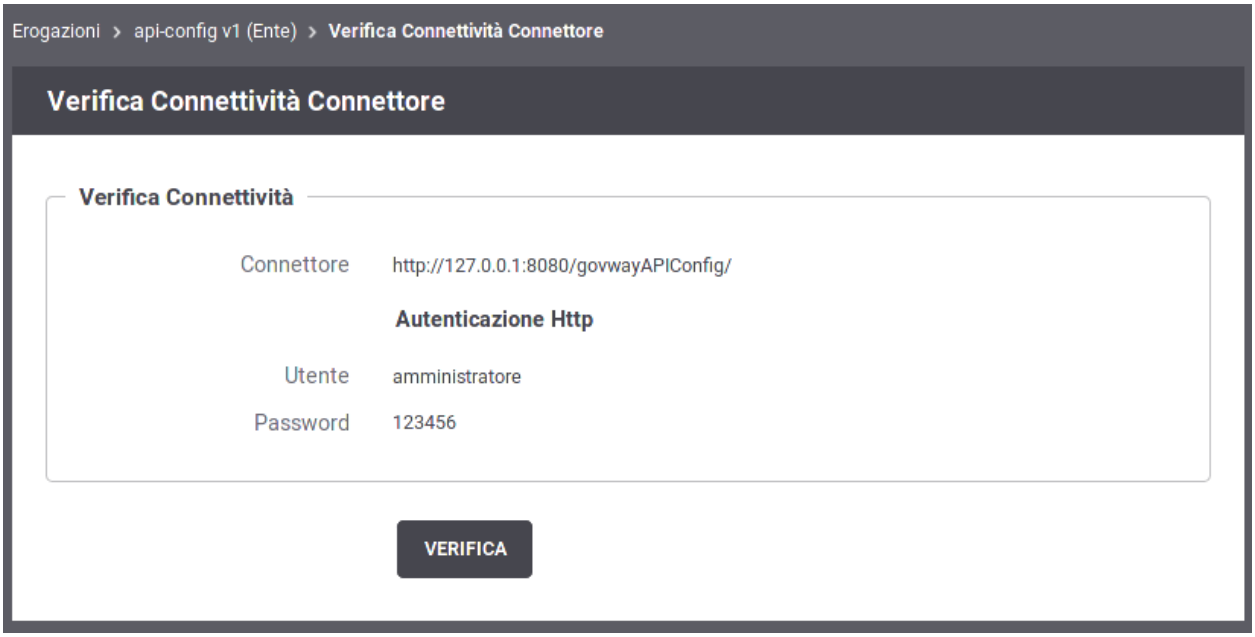


Fig. 2.23: Verifica del connettore

Dopo aver premuto il pulsante *Verifica* viene presentato l'esito della verifica di raggiungibilità (Fig. 2.24).

In presenza di un endpoint https, è possibile effettuare il download dei certificati ritornati dal server cliccando sul link "Download Certificati Server". Il formato del file scaricato è un PEM contenente tutti i certificati ritornati dal server.

2.7.2 Applicativi Server

Un applicativo di tipo "Server" consente di censire un'applicazione di backend alla quale associare quelle informazioni tipicamente indicate finora nella sezione "Connettore" dell'erogazione della API (endpoint, credenziali, . . .). In una erogazione è così possibile riferire un applicativo server già registrato come modalità alternativa a quella di indicare esplicitamente tutte le informazioni richieste.

Per registrare l'applicativo posizionarsi nella sezione *Registro* > *Applicativi*, quindi premere il pulsante *Aggiungi*. Compilare il form come segue (Fig. 2.26):

- *Profilo Interoperabilità*: Opzione visibile solo nel caso in cui non sia stata effettuata la relativa scelta sul menu della testata.

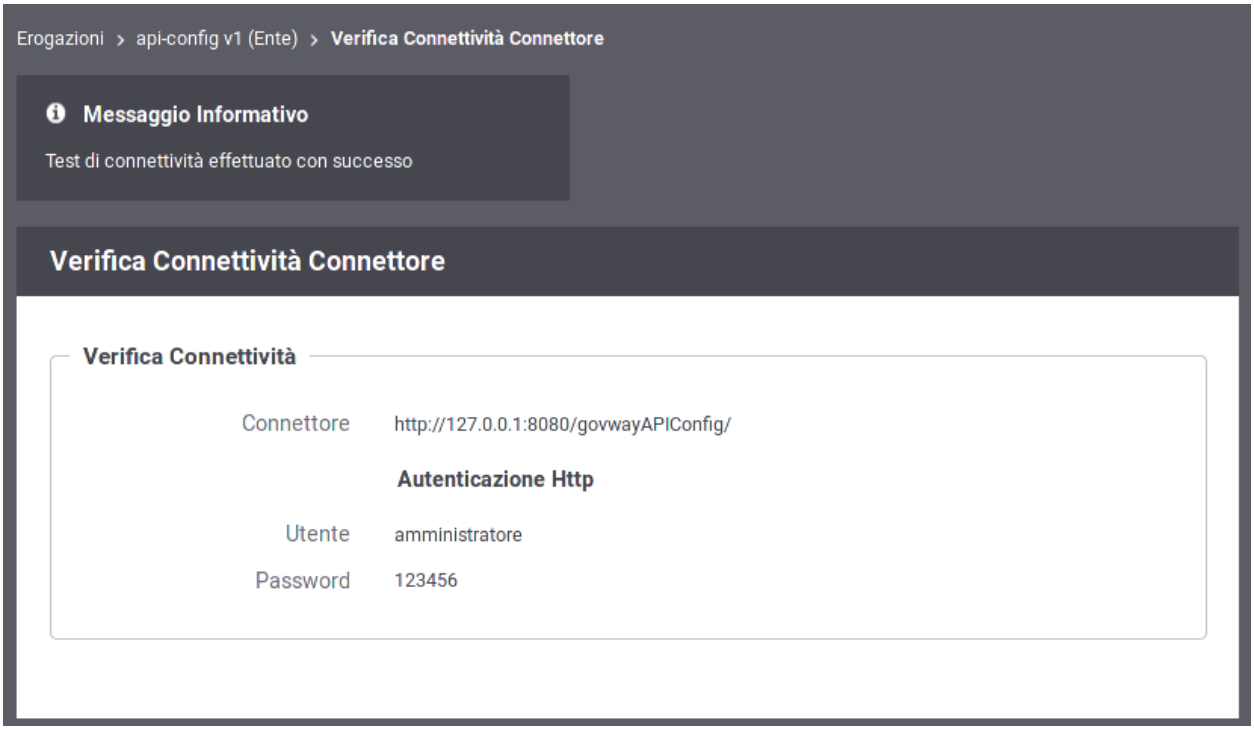


Fig. 2.24: Esito Verifica del connettore

- *Nome*: Assegnare un nome all'applicativo. È necessario che il nome indicato risulti univoco rispetto ai nomi già presenti per la modalità operativa selezionata (in questo caso API Gateway).
- *Tipo*: Utilizzare il tipo "Server" per censire un'applicativo di backend.
- *Connettore*: Tramite la sezione *Connettore* è possibile fornire quelle informazioni tipicamente indicate sinora nella sezione "Connettore" dell'erogazione di una API.

Dopo averlo creato, l'applicativo è associabile ad una Erogazione accedendo alla sezione «Connettore» come evinziato nella figura Fig. 2.27.

2.7.3 Load Balancer

Per le erogazioni di API è possibile definire connettori multipli con la finalità di attuare su di essi il bilanciamento delle richieste pervenute.

I contesti in cui l'interfaccia visualizza il valore di un connettore comprendono anche uno strumento per abilitare la gestione dei connettori multipli (Fig. 2.28).

Dopo aver premuto tale pulsante si accede ad una schermata che consente di abilitare e configurare tale funzionalità. In questa sezione, in particolare, viene descritta la funzionalità *Load Balancer* che rappresenta la voce di default una volta abilitato lo stato relativo alla configurazione dei connettori multipli (Fig. 2.29). Per maggiori dettagli si rimanda alla sezione *Configurazione del Bilanciamento del Carico*.

Una volta attivata la funzione di Load Balancer, nei contesti in cui l'interfaccia visualizzava l'endpoint di un connettore, viene adesso invece evidenziata la presenza di tale funzionalità. In tale contesto è possibile definire i nuovi connettori accedendo all'elenco dei connettori registrati tramite il nuovo pulsante dedicato (Fig. 2.30) come descritto nella sezione *Elenco dei Connettori Bilanciati*.

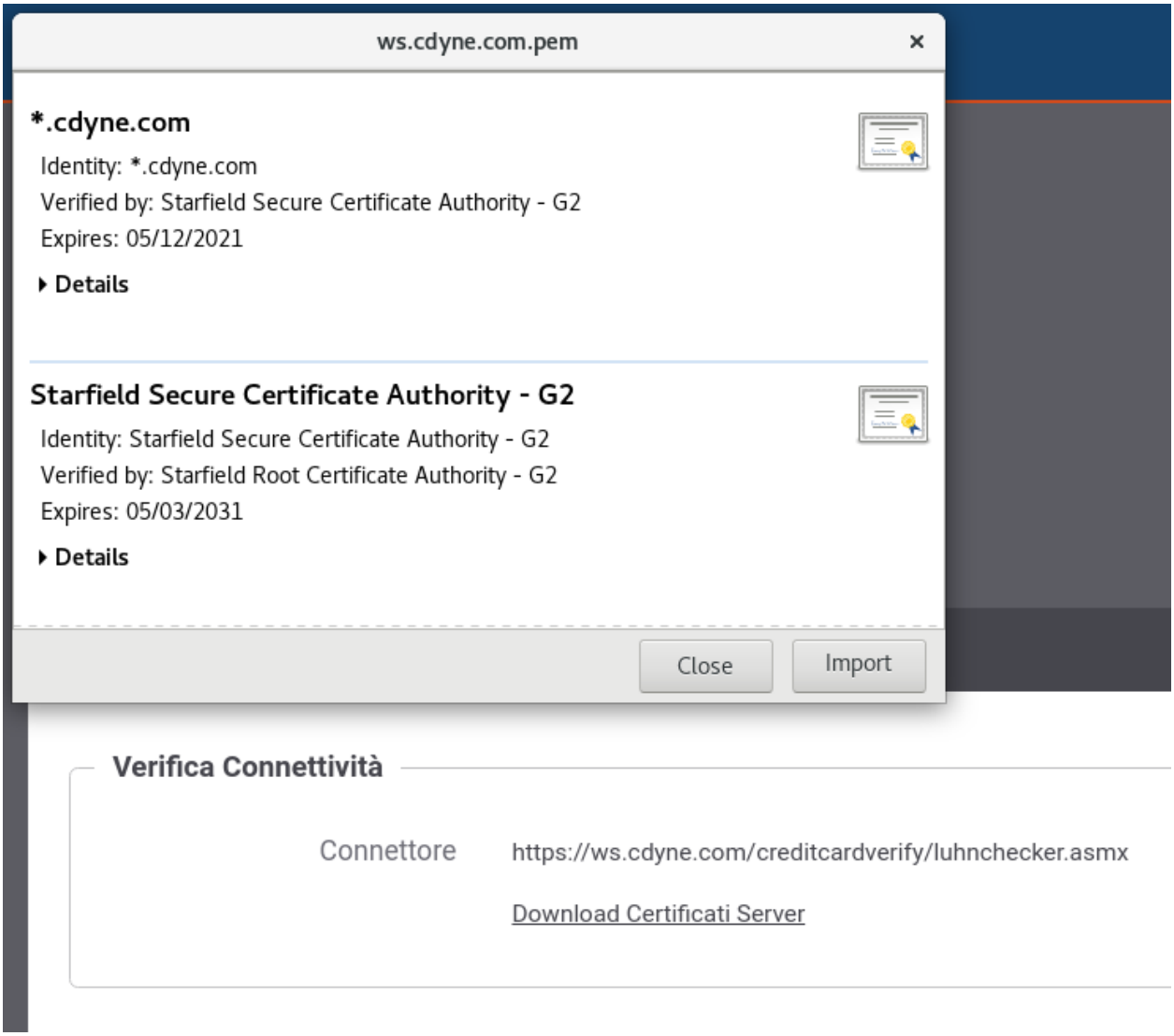


Fig. 2.25: Download Certificati Server

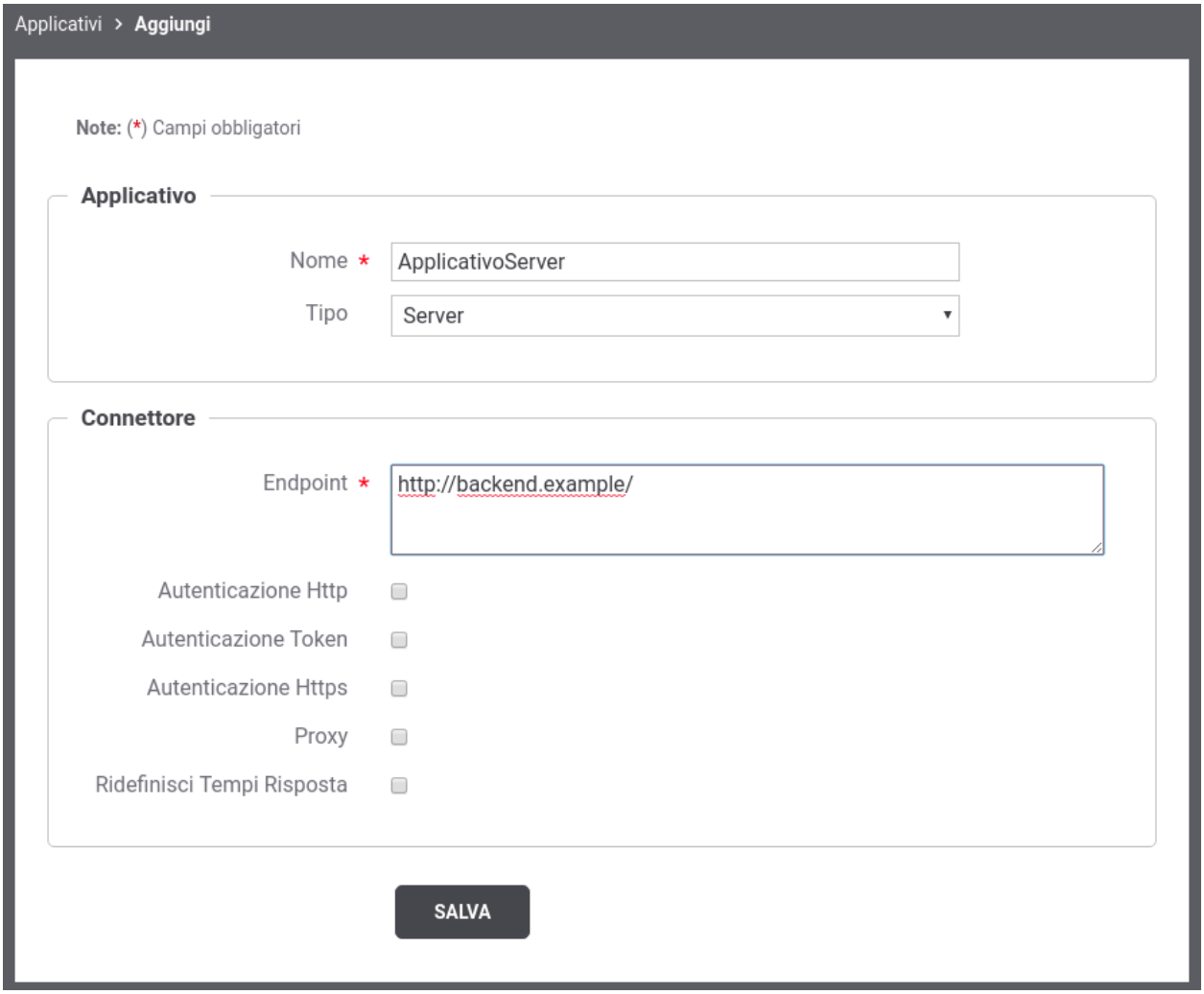


Fig. 2.26: Creazione di un Applicativo Server

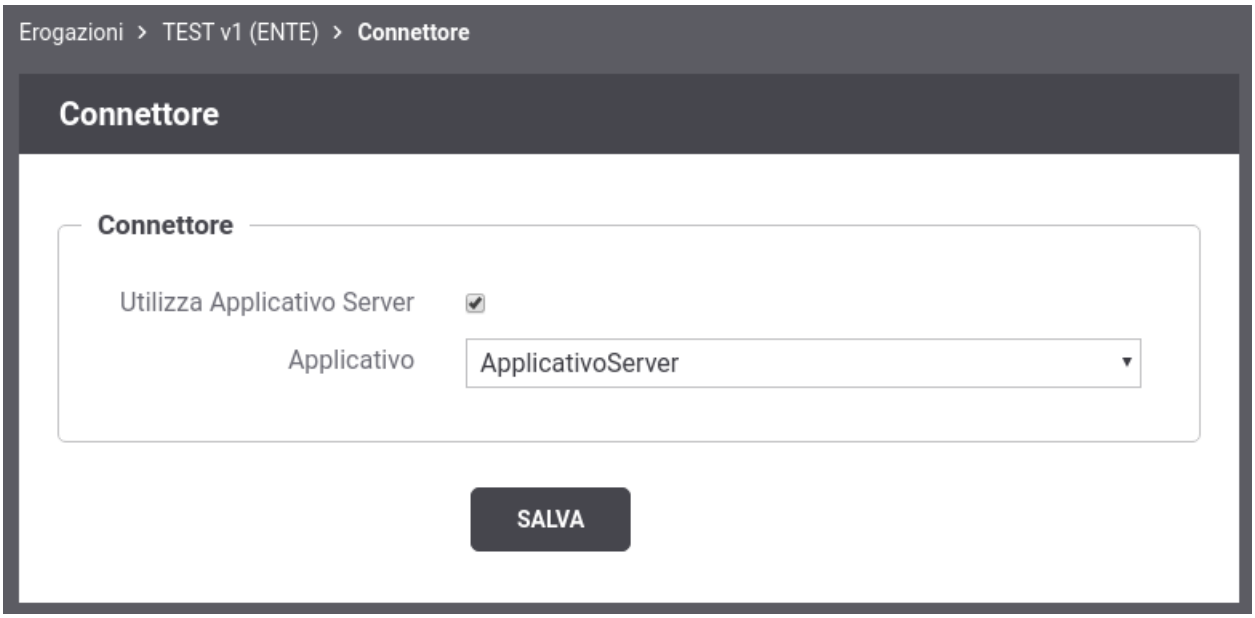


Fig. 2.27: Associazione di un Applicativo Server ad una Erogazione

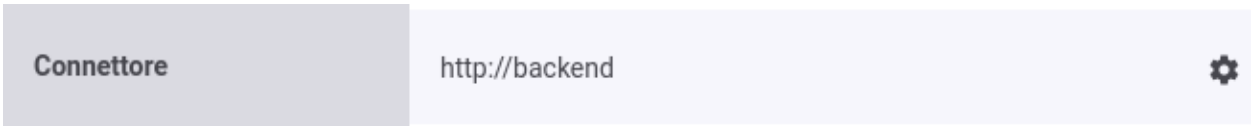


Fig. 2.28: Pulsante per la configurazione dei connettori multipli

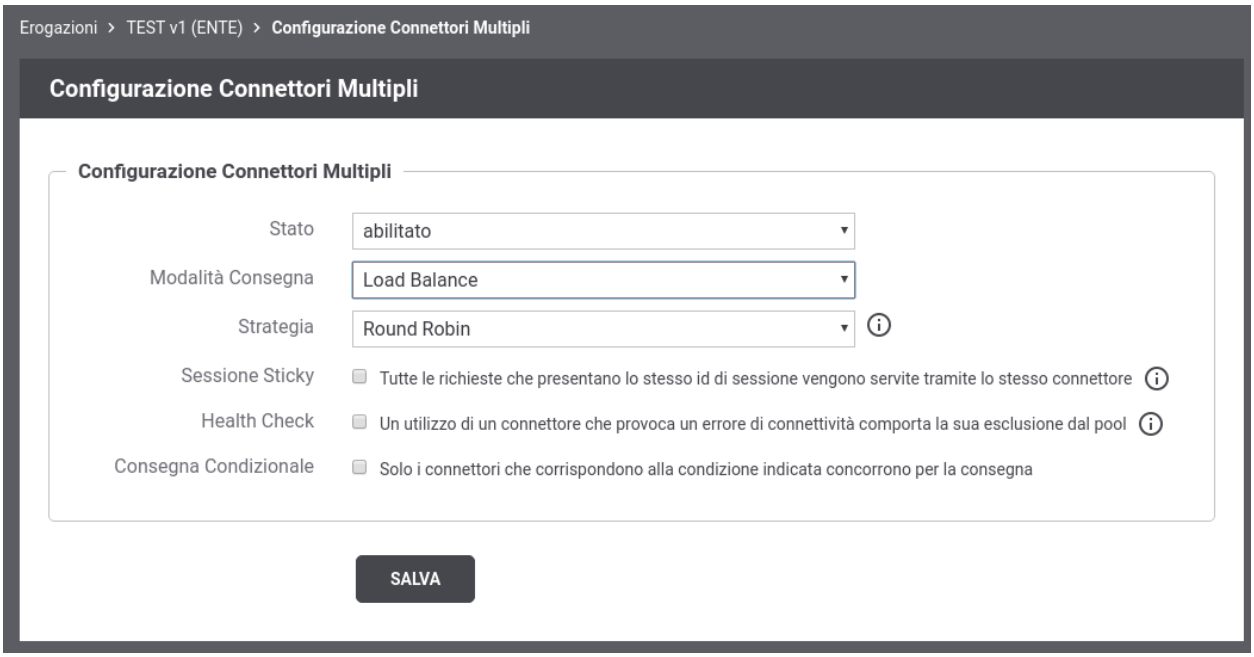


Fig. 2.29: Load Balancer

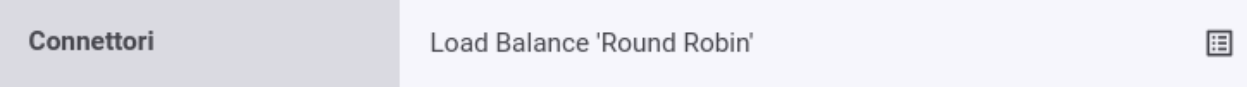


Fig. 2.30: Pulsante per accedere all'elenco dei connettori

Configurazione del Bilanciamento del Carico

Per abilitare la funzionalità di Load Balancer accedere alla sezione di dettaglio di una erogazione di API e cliccare sul pulsante di configurazione dei connettori multipli (Fig. 2.31).

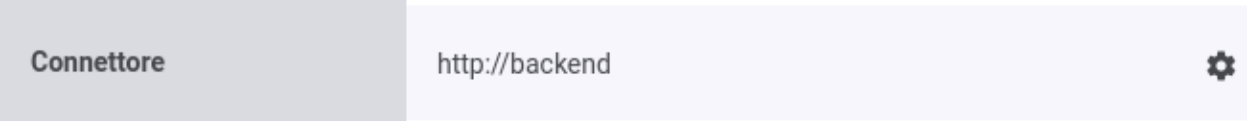


Fig. 2.31: Pulsante per la configurazione dei connettori multipli

Dopo aver premuto il pulsante si accede ad una schermata che consente di abilitare una funzionalità relativa ai connettori multipli. In questa sezione, in particolare, viene descritta la funzionalità *Load Balance* che rappresenta la voce di default una volta abilitato lo stato relativo alla configurazione dei connettori multipli (Fig. 2.32).

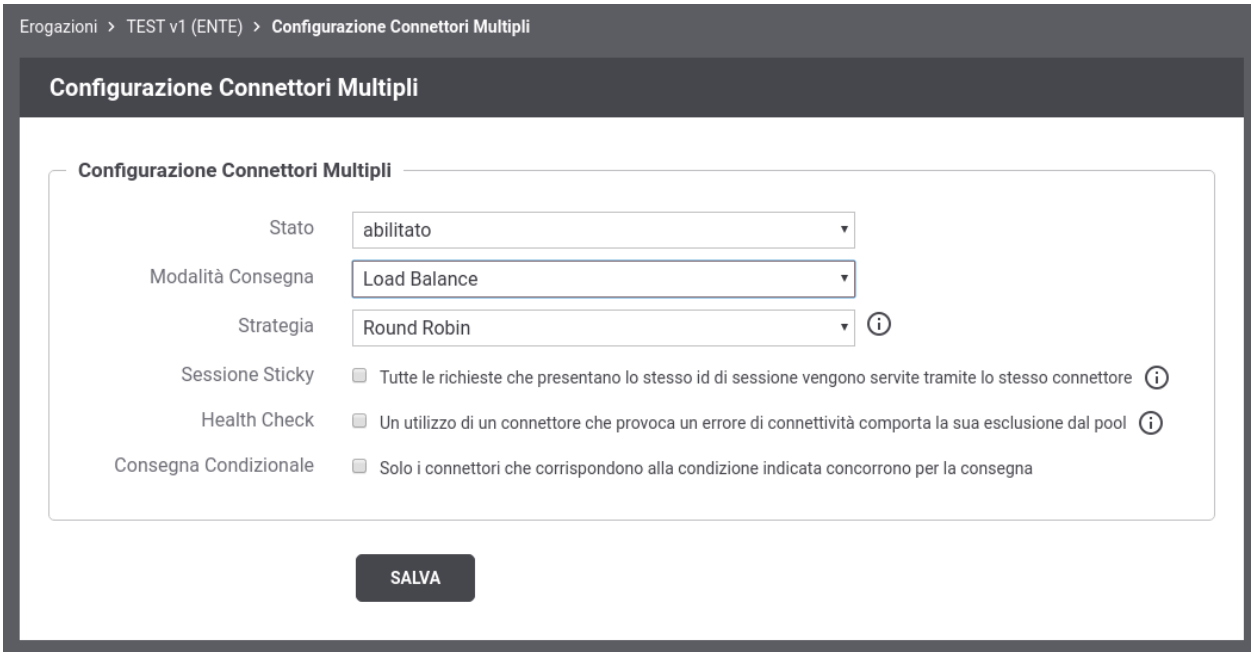


Fig. 2.32: Load Balancer

Vengono forniti differenti tipi di bilanciamento del carico:

- *Round Robin*: le richieste vengono distribuite in ordine tra i connettori registrati;
- *Weight Round Robin*: rispetto al Round Robin consente di riequilibrare eventuali server eterogenei tramite una distribuzione bilanciata rispetto al peso associato ad ogni connettore;
- *Random*: le richieste vengono distribuite casualmente tra i connettori registrati;
- *Weight Random*: rispetto al Random si ha una distribuzione casuale che considerà però il peso associato ad ogni connettore;

- *Source IP hash*: combina l'indirizzo IP del client e l'eventuale indirizzo IP portato in un header appartenente alla classe «Forwarded-For» o «Client-IP» per generare una chiave hash che viene designata per un connettore specifico;
- *Least Connections*: la richiesta viene indirizzata verso il connettore che ha il numero minimo di connessioni attive.

La configurazione permette anche di abilitare una sessione sticky in modo che tutte le richieste che presentano lo stesso id di sessione vengano servite tramite lo stesso connettore. Se l'identificativo di sessione si riferisce ad una nuova sessione, viene selezionato un connettore rispetto alla strategia indicata.

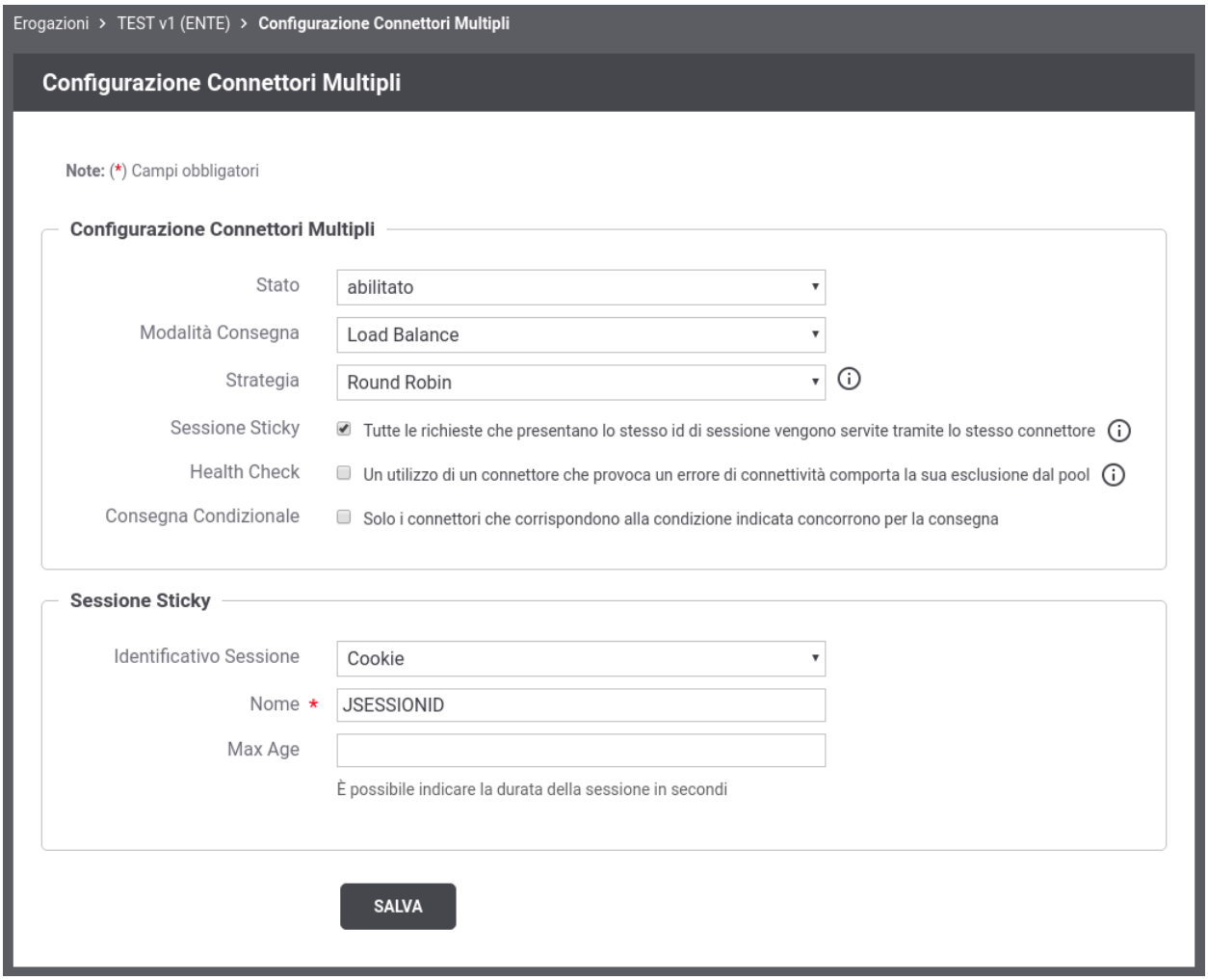


Fig. 2.33: Load Balancer con Sessione Sticky

L'identificativo di sessione utilizzato è individuabile tramite una delle seguenti modalità (Fig. 2.33):

- *Cookie*: nome di un cookie;
- *Header HTTP*: nome di un header http;
- *Url di Invocazione*: espressione regolare applicata sulla url di invocazione (l'espressione deve avere un match con l'intera url);
- *Parametro della Url*: nome del parametro presente nella url di invocazione;
- *Contenuto*: espressione (xPath o jsonPath) utilizzata per estrarre un identificativo dal body della richiesta;

- *Client IP*: indirizzo IP del client;
- *X-Forwarded-For*: header http appartenente alla classe «Forwarded-For» o «Client-IP»;
- *Template*: l'identificativo di sessione è il risultato dell'istanziazione del template fornito rispetto ai dati della richiesta;
- *Freemarker Template*: l'identificativo di sessione è ottenuto tramite il processamento di un Freemarker Template;
- *Velocity Template*: l'identificativo di sessione è ottenuto tramite il processamento di un Velocity Template;

È anche possibile attivare un "Passive Health Check" che verifica la connettività verso i connettori configurati. Un utilizzo di un connettore che provoca un errore di connettività comporta la sua esclusione dal pool dei connettori utilizzabili per un intervallo di tempo configurabile (Fig. 2.34).

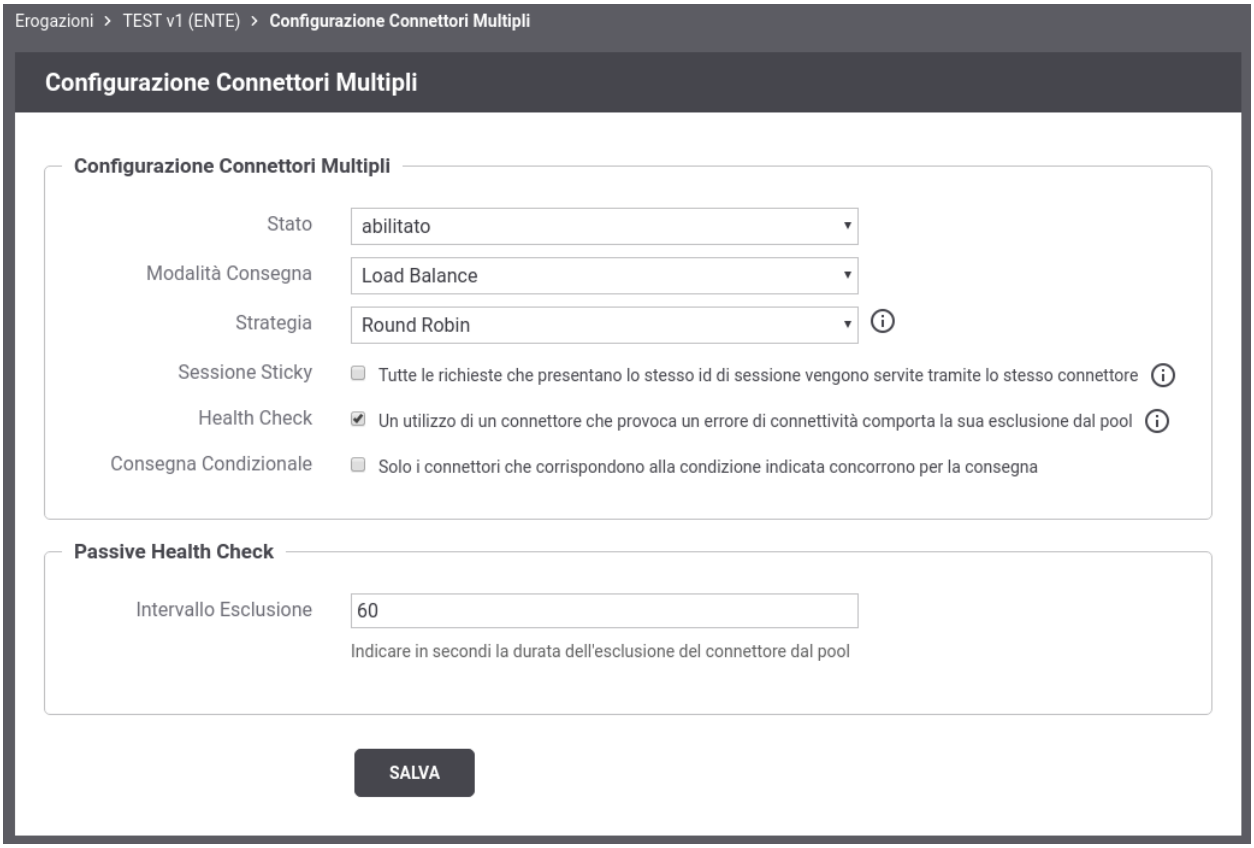


Fig. 2.34: Load Balancer con Passive Health Check

È infine possibile attivare una funzione di selezione dei connettori che partecipano al bilanciamento delle richieste in funzione di parametri della richiesta stessa (Fig. 2.35). Per ulteriori dettagli sulle modalità di selezione condizionale dei connettori si rimanda alla sezione *Consegna Condizionale* poichè gli aspetti di questa configurazione sono identici a quelli descritti per la funzionalità di consegna condizionale.

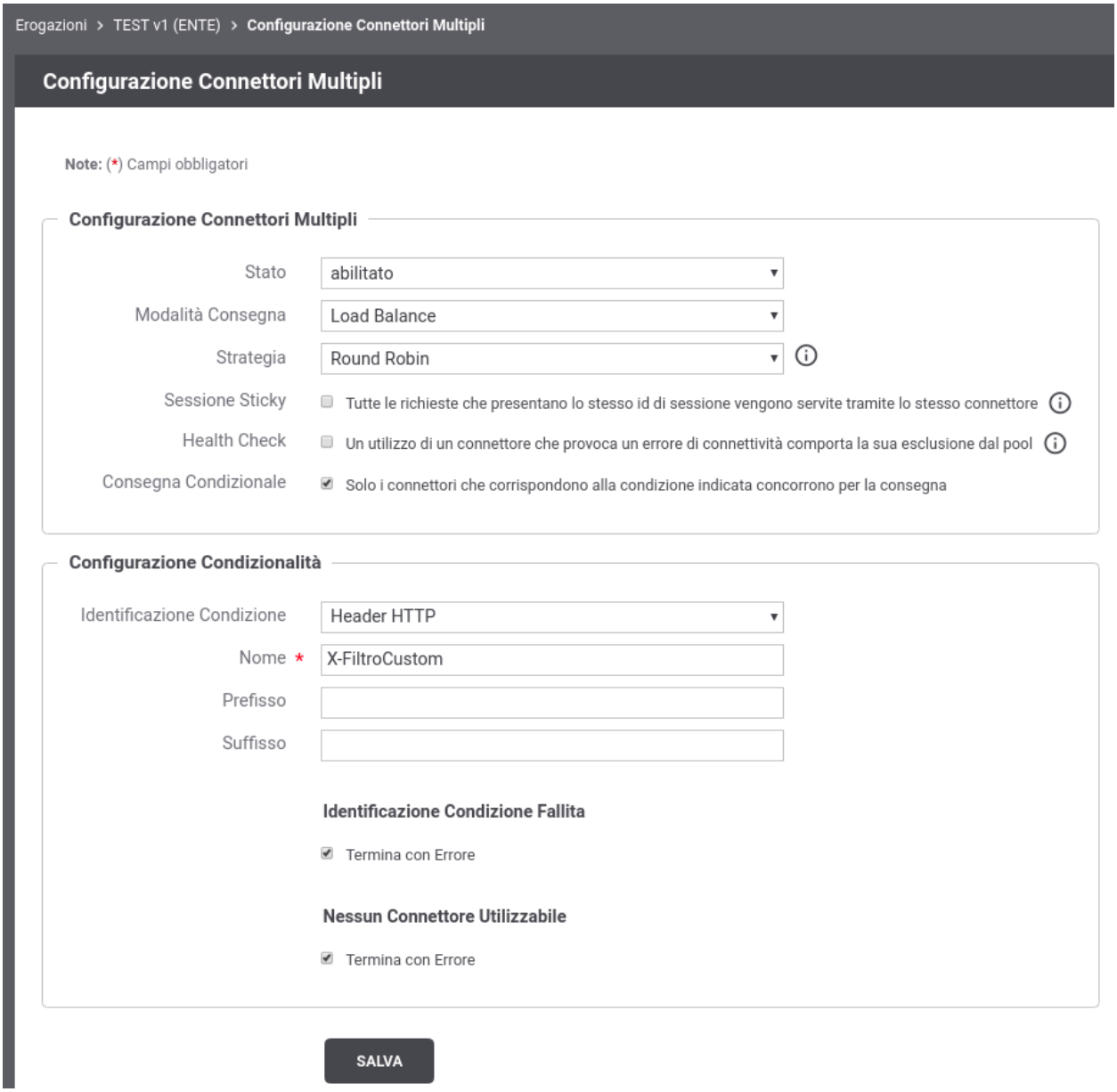


Fig. 2.35: Selezione condizionale dei connettori che partecipano al bilanciamento

Elenco dei Connettori Bilanciati

Per le erogazioni di API è possibile definire connettori multipli con finalità di bilanciamento delle richieste in arrivo.

Dopo aver attivato la funzione di Load Balancer, nei contesti in cui l'interfaccia visualizzava l'endpoint di un connettore, viene adesso invece evidenziata la presenza della funzionalità di Load Balancer. In tale contesto è possibile definire i nuovi connettori accedendo all'elenco dei connettori registrati tramite il nuovo pulsante dedicato (Fig. 2.36).

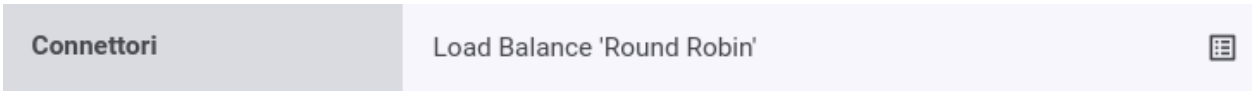


Fig. 2.36: Pulsante per accedere all'elenco dei connettori

Accedendo all'elenco la prima volta si troverà il solo connettore di default definito al momento della registrazione dell'API erogata (Fig. 2.37).

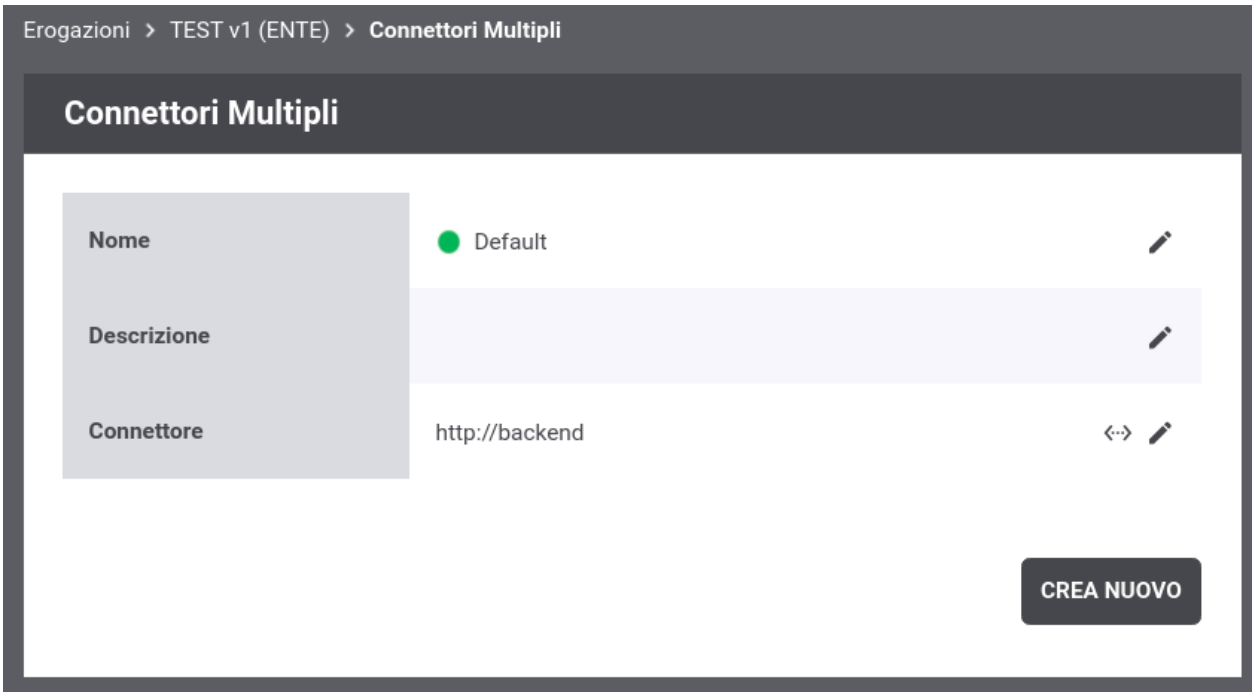


Fig. 2.37: Elenco dei connettori bilanciati con presenza del solo connettore di default

Tramite il pulsante *Crea Nuovo* è possibile registrare un nuovo connettore. Compilare il form come segue (Fig. 2.38):

- *Nome*: Assegnare un nome al connettore. È necessario che il nome indicato risulti univoco all'interno del pool dei connettori definiti per l'API.
- *Stato*: Indica lo stato del connettore. È possibile abilitare o disabilitare il singolo connettore anche dopo che è stato definito.
- *Descrizione*: Permette di fornire una descrizione generica.
- *Connettore*: Tramite la sezione *Connettore* è possibile fornire quelle informazioni tipicamente indicate sinora nella sezione "Connettore" dell'erogazione di una API.

All'interno della definizione dei dati di un connettore, è anche possibile riferire un Applicativo di tipo "server" precedentemente registrato come descritto nella sezione *Applicativi Server* (Fig. 2.39).

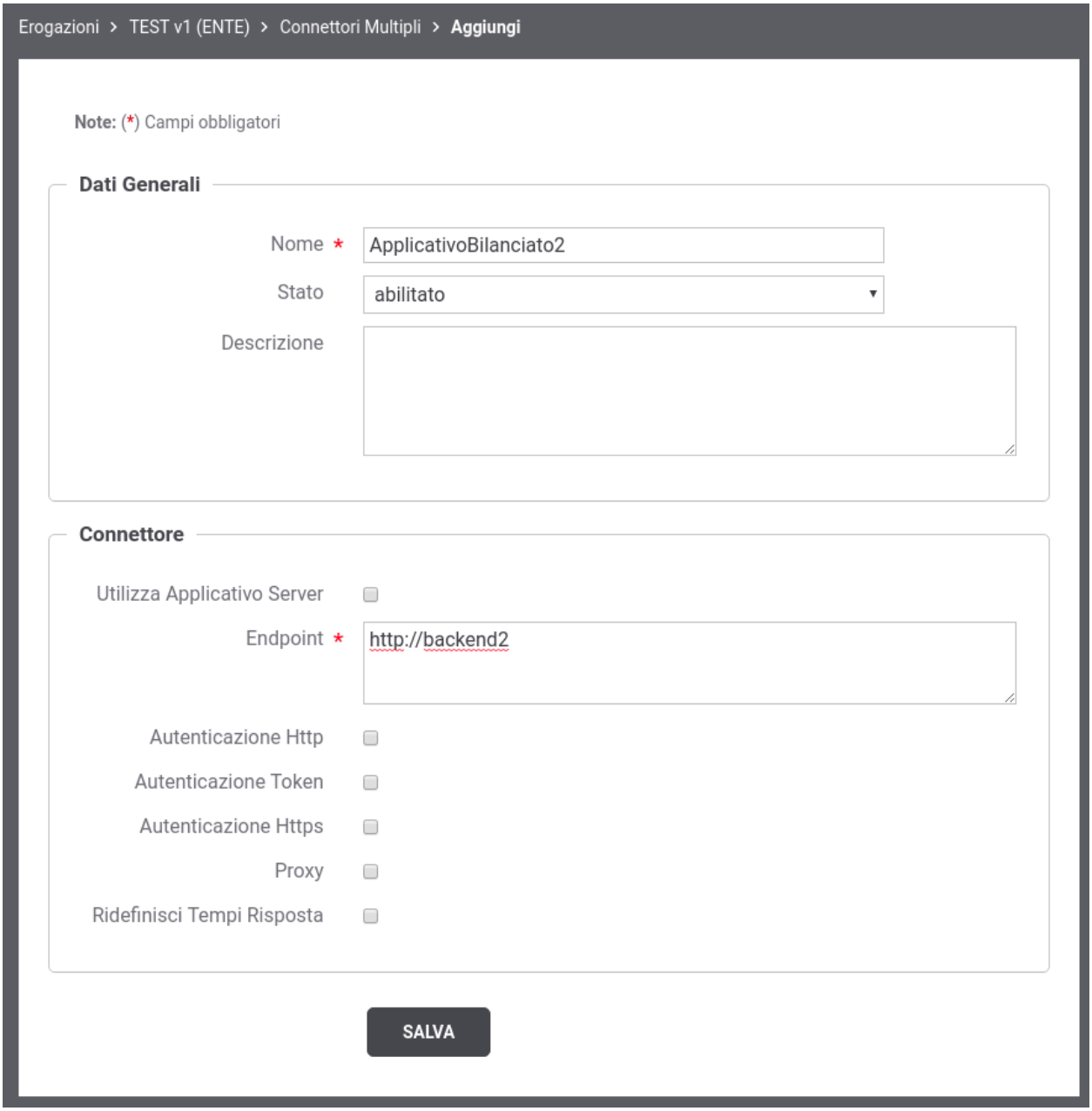


Fig. 2.38: Registrazione di un nuovo connettore per il bilanciamento del carico

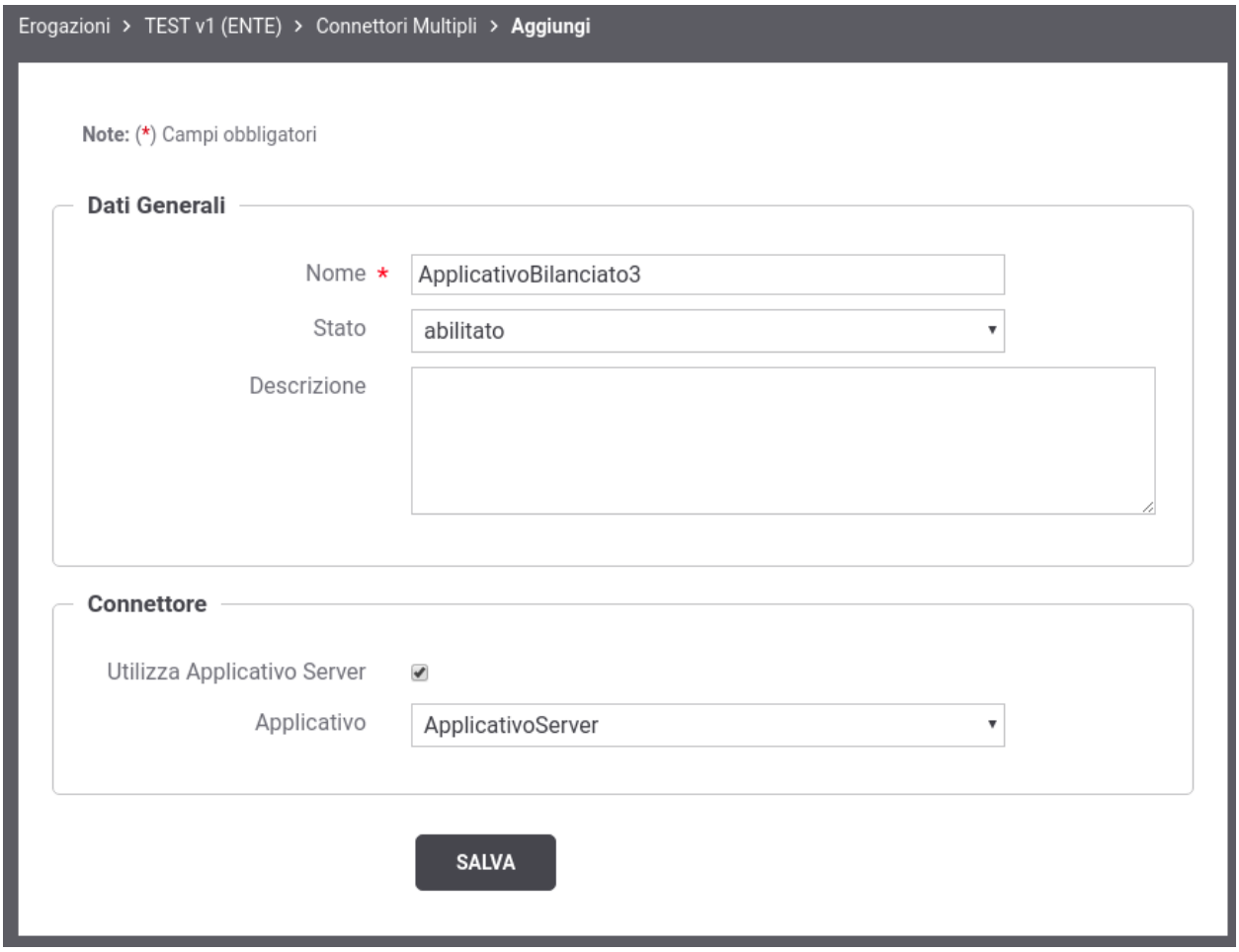


Fig. 2.39: Registrazione di un nuovo connettore, per il bilanciamento del carico, che riferisce un Applicativo Server

I nuovi connettori creati sono accessibili nell'elenco dei connettori (Fig. 2.40). I tab presenti nell'elenco riportano i nomi dei connettori configurati, e selezionando quello di interesse è possibile visualizzare e/o modificare i dati del connettore oltre ad eliminarlo tramite il pulsante *Elimina*.

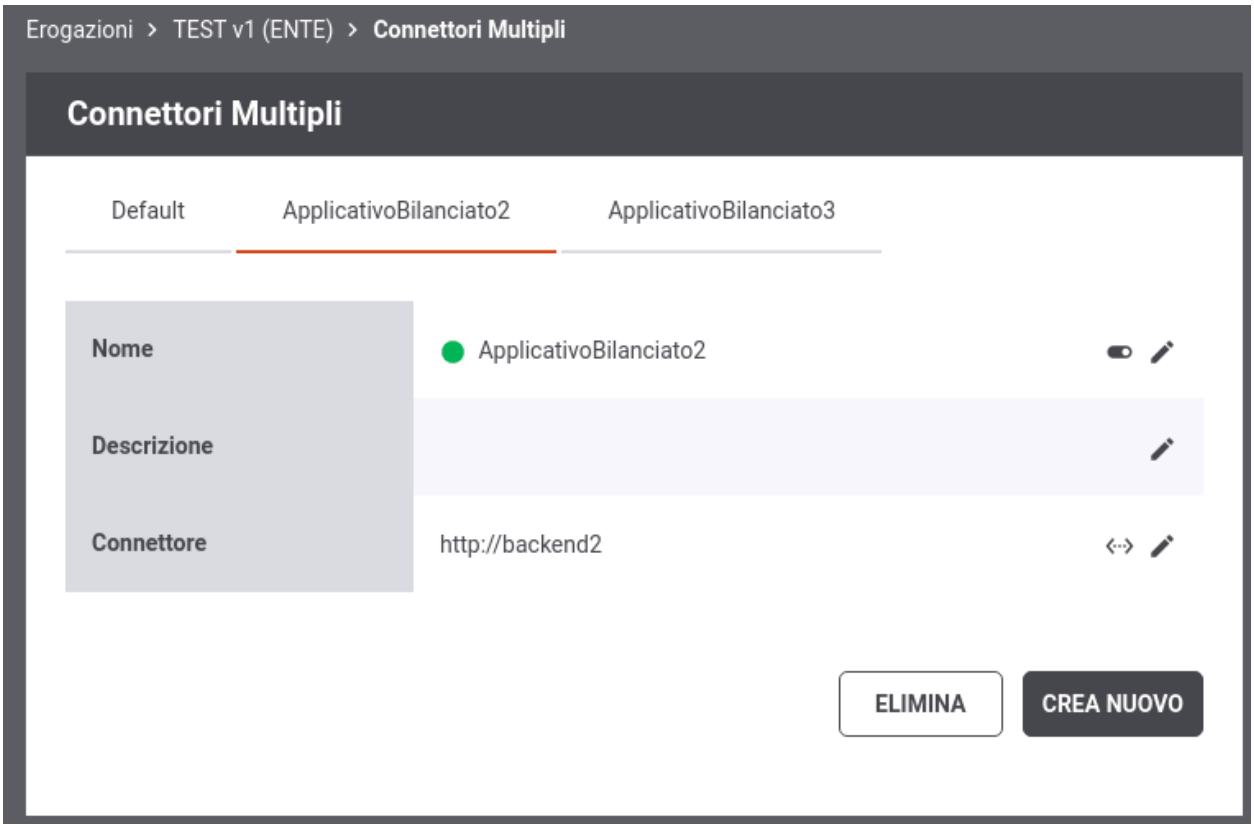


Fig. 2.40: Elenco dei connettori bilanciati

Nel caso sia stato selezionato un tipo di Load Balancer "Weight" nell'elenco dei connettori sarà possibile anche associare un peso maggiore al singolo connettore (Fig. 2.41).

2.7.4 Consegna Condizionale

Per le erogazioni di API è possibile definire connettori diversi, selezionati dinamicamente al verificarsi di specifiche condizioni.

Per abilitare una consegna condizionale è necessario accedere al dettaglio di una erogazione, dove viene visualizzato il valore del connettore, e successivamente cliccare sul pulsante che consente di gestire la configurazione dei connettori multipli (Fig. 2.42).

Dopo aver premuto il pulsante si accede ad una schermata che consente di abilitare una funzionalità relativa ai connettori multipli. In questa sezione, in particolare, viene descritta la funzionalità *Consegna Condizionale* che dovrà essere selezionata una volta abilitato lo stato relativo alla configurazione dei connettori multipli (Fig. 2.43). Per maggiori dettagli si rimanda alla sezione *Configurazione della Consegna Condizionale*.

Una volta attivata la funzione di Consegna Condizionale, nei contesti in cui l'interfaccia visualizzava l'endpoint di un connettore, viene adesso invece evidenziata la presenza di tale funzionalità. In tale contesto è possibile definire i nuovi connettori accedendo all'elenco dei connettori registrati tramite il nuovo pulsante dedicato (Fig. 2.44) come descritto nella sezione *Elenco dei Connettori*.

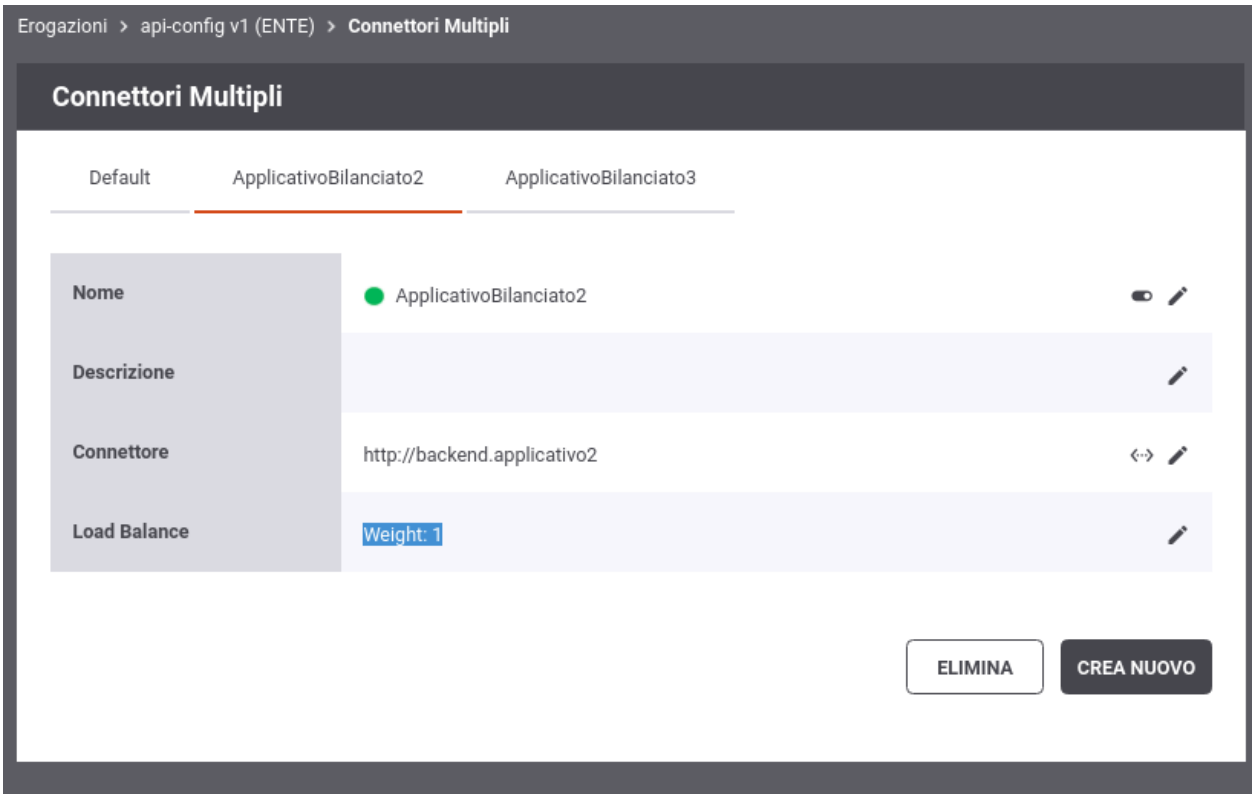


Fig. 2.41: Elenco dei connettori bilanciati con opzione “Weight”

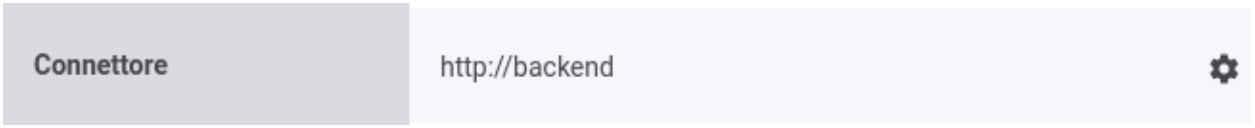


Fig. 2.42: Pulsante per la configurazione dei connettori multipli

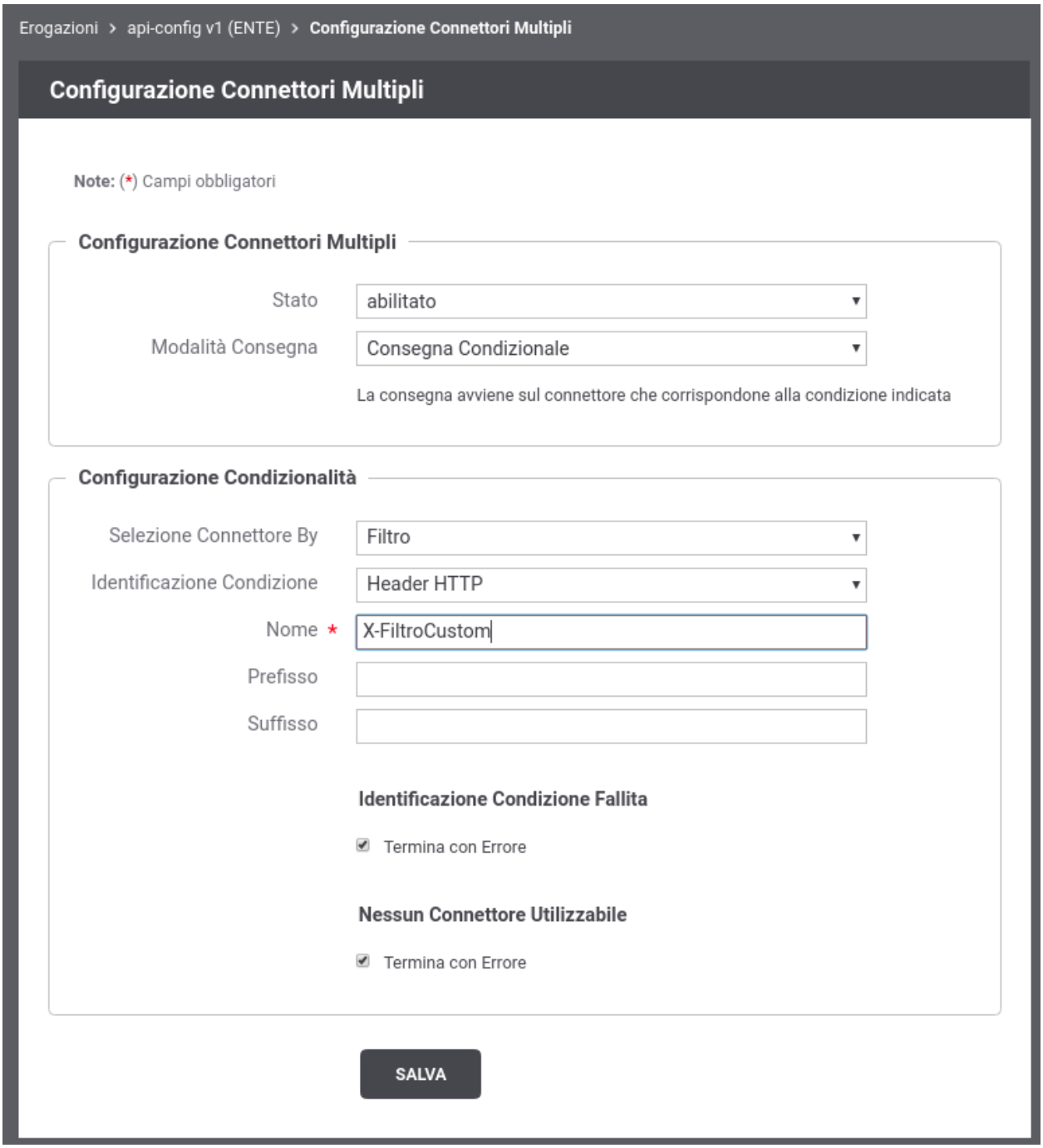


Fig. 2.43: Consegna Condizionale

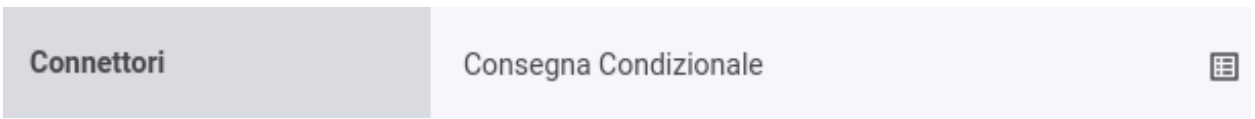


Fig. 2.44: Pulsante per accedere all'elenco dei connettori

Configurazione della Consegna Condizionale

Per abilitare la funzionalità di Consegna Condizionale accedere alla sezione di dettaglio di una erogazione di API e cliccare sul pulsante di configurazione dei connettori multipli (Fig. 2.45).

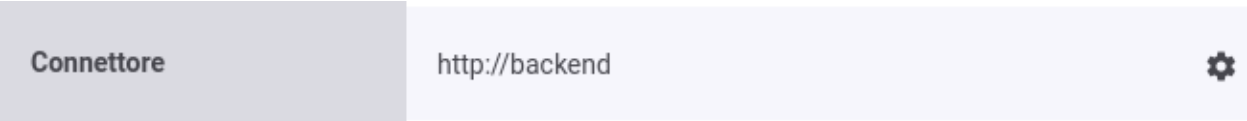


Fig. 2.45: Pulsante per la configurazione dei connettori multipli

Dopo aver premuto il pulsante si accede ad una schermata che consente di abilitare una funzionalità relativa ai connettori multipli. In questa sezione, in particolare, viene descritta la funzionalità *Consegna Condizionale* che dovrà essere selezionata una volta abilitato lo stato relativo alla configurazione dei connettori multipli (Fig. 2.46).

Il connettore su cui verrà inoltrata la richiesta pervenuta sul Gateway, viene selezionato in base al suo nome o a un filtro associato al connettore stesso. La modalità di selezione desiderata deve essere indicata tramite la voce "Selezione Connettore By". Il valore del filtro (utilizzato per identificare il connettore di consegna) o il nome del connettore stesso, viene individuato all'interno della richiesta attraverso una delle seguenti modalità:

- *Header HTTP*: nome di un header http;
- *Url di Invocazione*: espressione regolare applicata sulla url di invocazione (l'espressione deve avere un match con l'intera url);
- *Parametro della Url*: nome del parametro presente nella url di invocazione;
- *SOAPAction*: individua una operazione SOAP;
- *Contenuto*: espressione (xPath o jsonPath) utilizzata per estrarre un identificativo dal body della richiesta;
- *Client IP*: indirizzo IP del client;
- *X-Forwarded-For*: header http appartenente alla classe «Forwarded-For» o «Client-IP»;
- *Template*: l'identificativo di sessione è il risultato dell'istanziazione del template fornito rispetto ai dati della richiesta;
- *Freemarker Template*: l'identificativo di sessione è ottenuto tramite il processamento di un Freemarker Template;
- *Velocity Template*: l'identificativo di sessione è ottenuto tramite il processamento di un Velocity Template;

I campi "Prefisso" e "Suffisso" permettono di anteporre al valore estratto dalla richiesta un prefisso e/o un suffisso prima di utilizzare tale valore per l'identificazione del connettore (sia tramite nome che tramite filtro).

Tramite le checkbox "Termina con Errore" è infine possibile configurare l'erogazione per utilizzare uno specifico connettore di default, invece di terminare la transazione con errore, nel caso la condizione non sia presente nella richiesta o non permetta di identificare alcun connettore all'interno del pool. Nel caso non venga terminata la transazione con errore, è anche possibile impostare l'emissione o meno di un diagnostico che segnali la condizione fallita (esempio riportato nella figura Fig. 2.47).

Le regole per la selezione del connettore sopra descritte possono essere ridefinite per singole o gruppi di operazioni attraverso la definizione di regole specifiche, accedendo al link "regole" presente nella maschera di configurazione.

La creazione di una regola specifica deve innanzitutto identificare le operazioni dell'API a cui la regola è riferita tramite il campo "Risorsa" o "Azione" attraverso una delle seguenti modalità:

- *Nome Azione o Risorsa*: il nome esatto dell'azione o della risorsa su cui verrà applicativa la regola; può in alternativa essere utilizzata un'espressione regolare (es. ^(?:POST.operazione1|GET.operazione2)$)

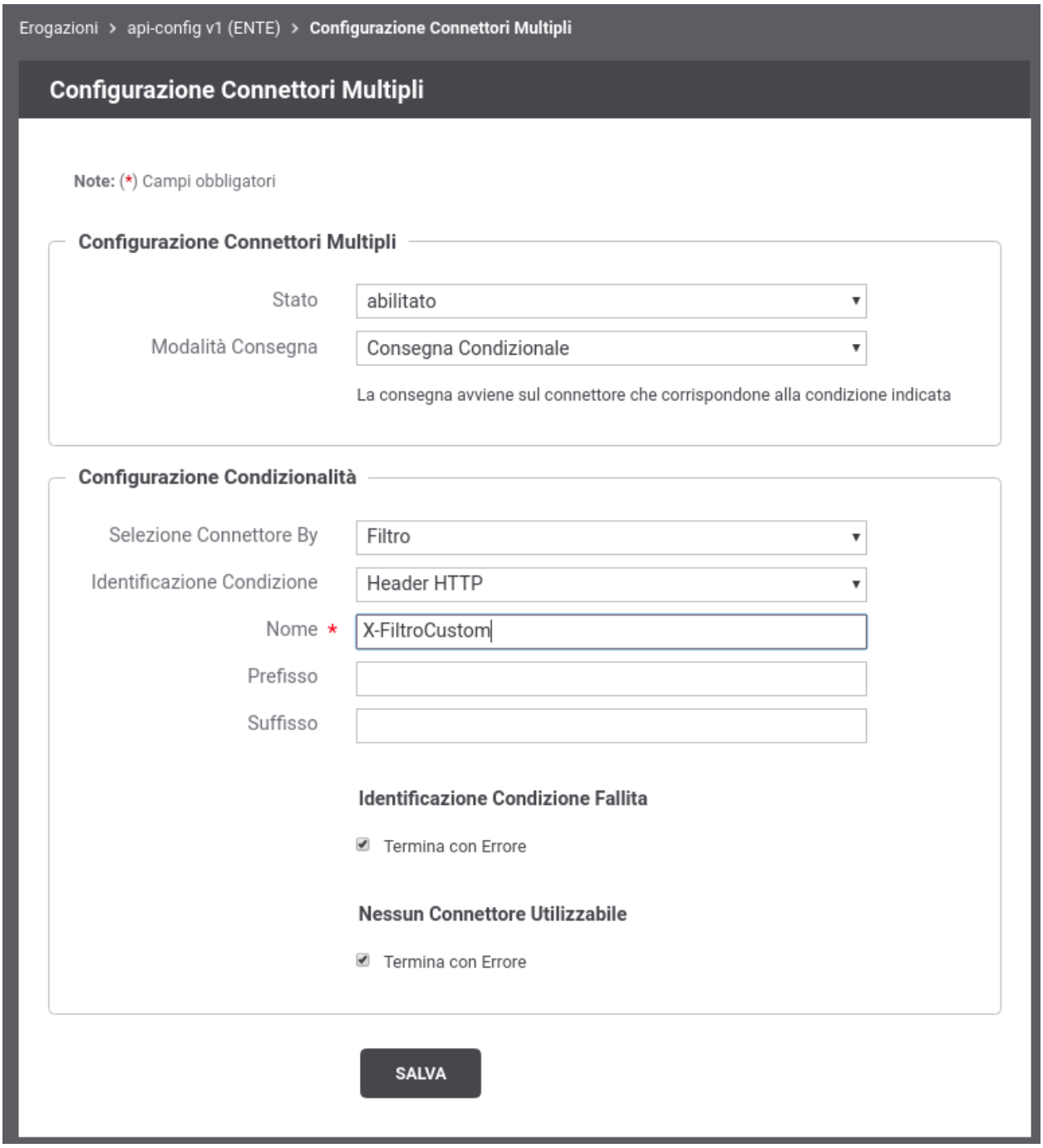


Fig. 2.46: Consegna Condizionale

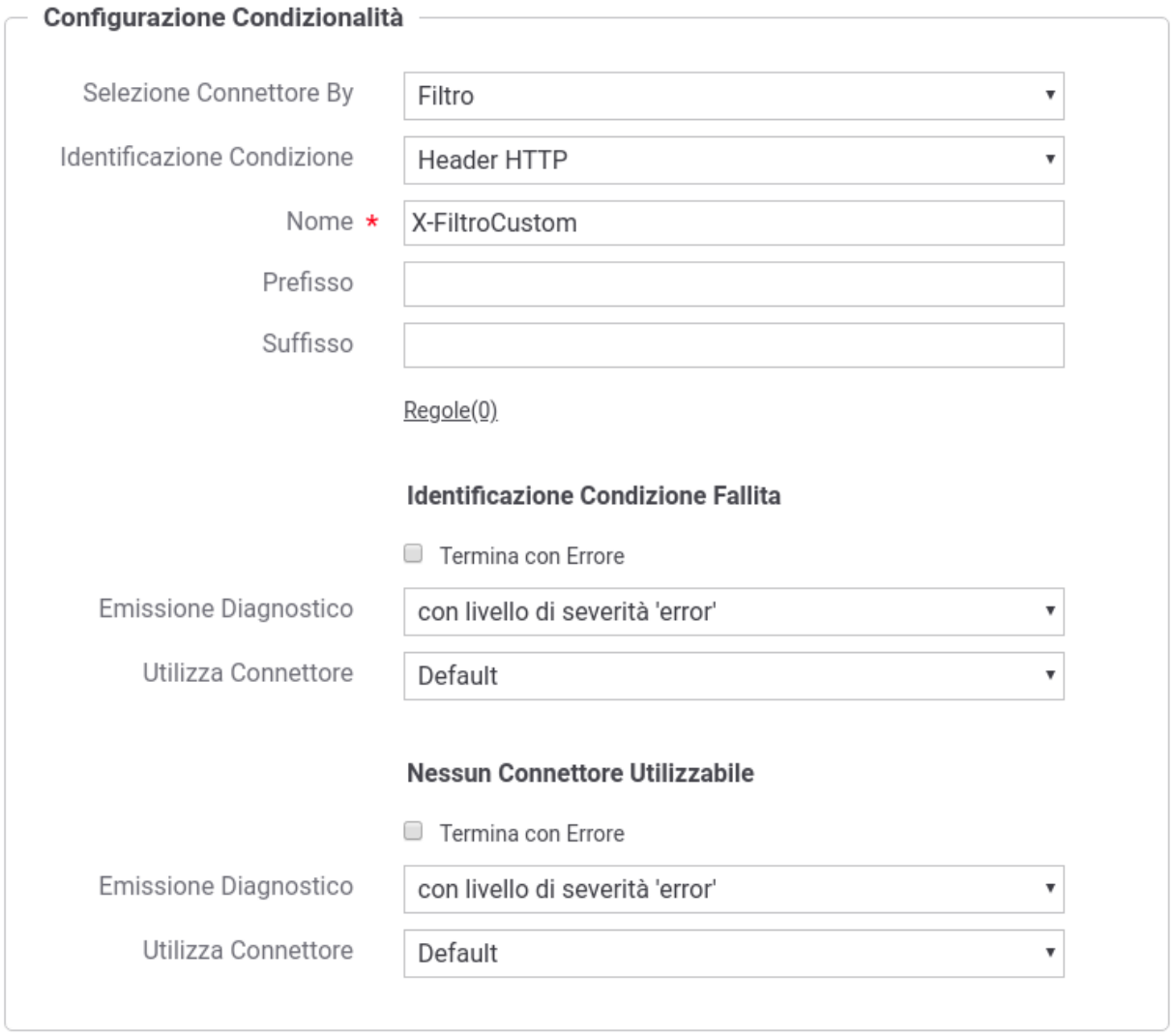


Fig. 2.47: Consegna Condizionale con Connettore di Default

- *HttpMethod e Path* (utilizzabile solo su API REST): metodo http e path di una risorsa dell'API; è possibile indicare qualsiasi metodo o qualsiasi path con il carattere speciale "*". È inoltre possibile definire solamente la parte iniziale di un path attraverso lo "*". Alcuni esempi:
  - "POST /resource"
  - "* /resource"
  - "POST *"
  - "* /resource/*"

Nella figura Fig. 2.48 viene visualizzata la maschera di creazione di una regola specifica. Le modalità di identificazione del nome del connettore o del valore del filtro sono le medesime descritte in precedenza.

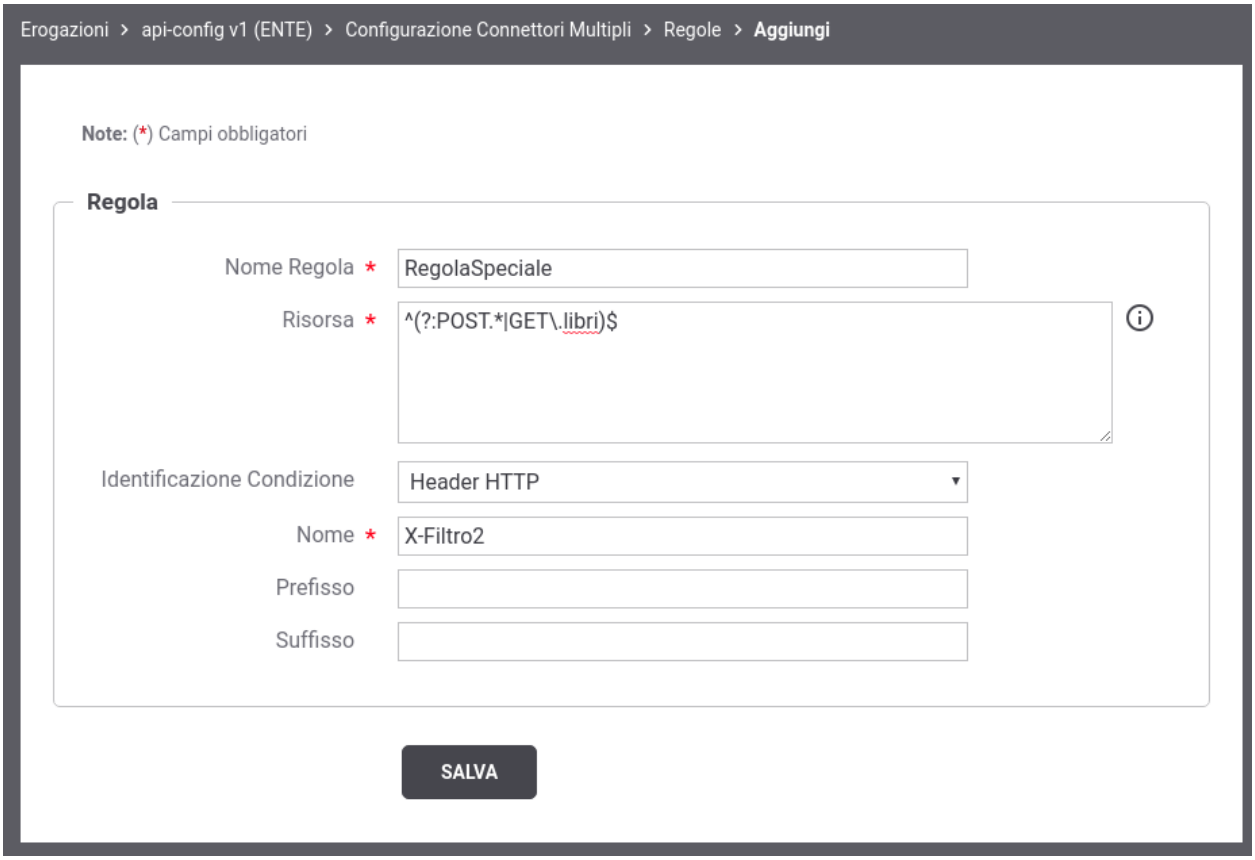


Fig. 2.48: Regola di Consegna Condizionale per Operazione

Elenco dei Connettori

Per le erogazioni di API è possibile definire connettori multipli con finalità di selezione condizionale del connettore a cui inoltrare le richieste in arrivo.

Dopo aver attivato la funzione di Consegna Condizionale, nei contesti in cui l'interfaccia visualizzava l'endpoint di un connettore, viene adesso invece evidenziata la presenza della funzionalità attivata. In tale contesto è possibile definire i nuovi connettori accedendo all'elenco dei connettori registrati tramite il nuovo pulsante dedicato (Fig. 2.49).

Accedendo all'elenco la prima volta si troverà il solo connettore di default definito al momento della registrazione dell'API erogata (Fig. 2.50).

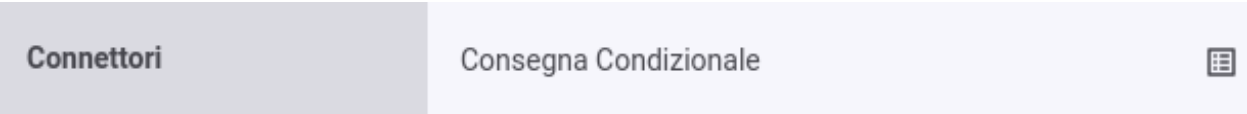


Fig. 2.49: Pulsante per accedere all'elenco dei connettori

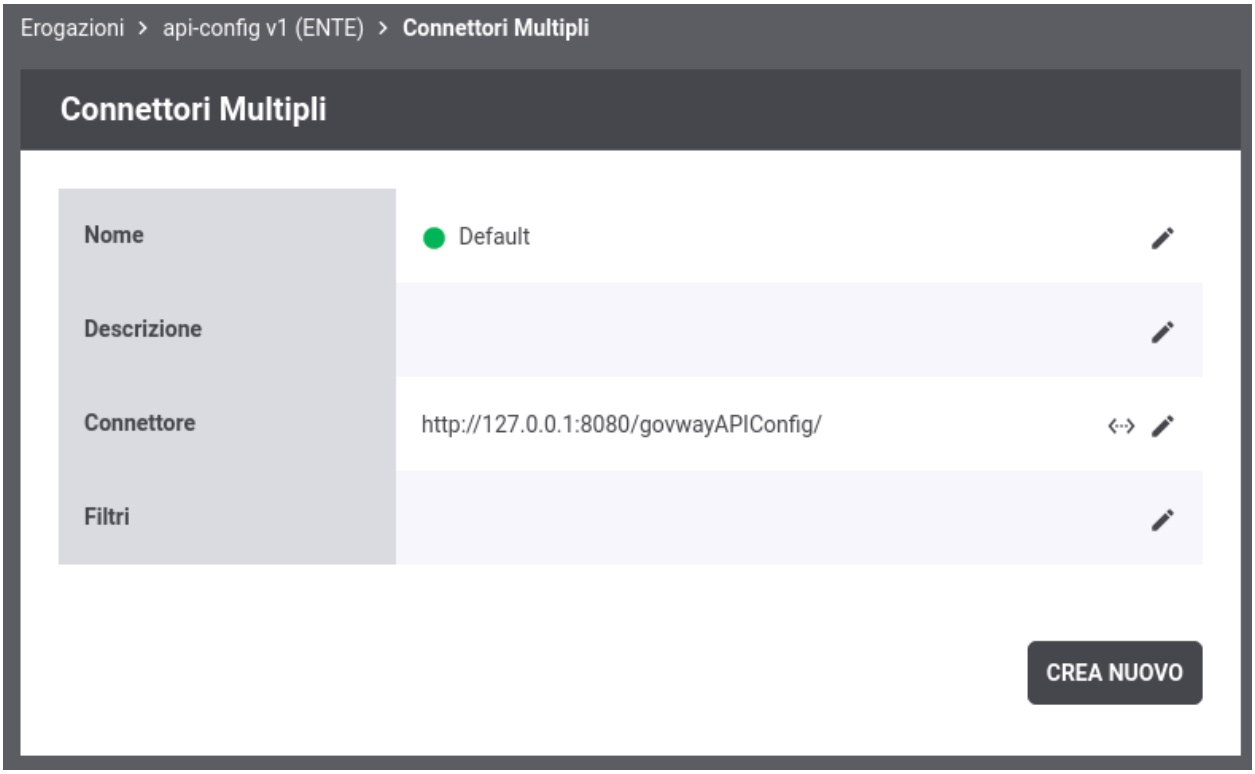


Fig. 2.50: Elenco dei connettori con presenza del solo connettore di default

Tramite il pulsante *Crea Nuovo* è possibile registrare un nuovo connettore. Compilare il form come segue (Fig. 2.51):

- *Nome*: Assegnare un nome al connettore. È necessario che il nome indicato risulti univoco all'interno del pool dei connettori definiti per l'API.
- *Stato*: Indica lo stato del connettore. È possibile abilitare o disabilitare il singolo connettore anche dopo che è stato definito.
- *Descrizione*: Permette di fornire una descrizione generica.
- *Filtri*: Nel caso sia stata configurata una selezione del connettore basata sui filtri, questo campo permette di assegnare al connettore i valori con cui verrà selezionato dal Gateway.
- *Connettore*: Tramite la sezione *Connettore* è possibile fornire quelle informazioni tipicamente indicate sinora nella sezione "Connettore" dell'erogazione di una API.

All'interno della definizione dei dati di un connettore, è anche possibile riferire un Applicativo di tipo "server" precedentemente registrato come descritto nella sezione *Applicativi Server* (Fig. 2.52).

I nuovi connettori creati sono accessibili nell'elenco dei connettori (Fig. 2.53). I tab presenti nell'elenco riportano i nomi dei connettori configurati, e selezionando quello di interesse è possibile visualizzare e/o modificare i dati del connettore oltre ad eliminarlo tramite il pulsante *Elimina*.

Nel caso sia stata configurata una selezione del connettore basata sui filtri, si deve procedere ad assegnare anche al connettore di default uno o più valori nei filtri in modo che sia selezionabile dal Gateway. Tale operazione non è necessaria solamente se si desidera utilizzare il connettore di default solamente nei casi in cui la condizione non è identificata nella richiesta o non abbia consentito ad identificare un connettore.

2.8 Gestione CORS

Quando un'applicazione client in esecuzione su un browser (es. codice javascript) richiede l'accesso ad una risorsa di un differente dominio, protocollo o porta tale richiesta viene gestita dal browser tramite una politica di *cross-origin HTTP request (CORS)*. Il CORS definisce un modo nel quale un browser ed un server (o il gateway) possono interagire per abilitare interazioni attraverso differenti domini.

In GovWay è possibile abilitare la gestione del CORS sia globalmente, in modo che sia valida per tutte le APIs, che singolarmente sulla singola erogazione o fruizione.

È possibile modificare le impostazioni CORS seguendo il collegamento presente nella riga *Gestione CORS* del dettaglio di una erogazione o fruizione. L'impostazione permette di ridefinire la configurazione globale; i campi del form sono i medesimi descritti nella configurazione globale (sezione *Gestione CORS*).

2.9 Differenziare le configurazioni specifiche per risorsa/azione

Le configurazioni specifiche che andiamo a descrivere si possono differenziare per sottoinsiemi delle azioni/risorse presenti nel servizio erogato/fruito. Il sistema crea automaticamente una configurazione unica, valida per tutte le azioni/risorse del servizio. Per intervenire su tale configurazione, o crearne di nuove, sia accede al collegamento presente nella colonna *Configurazione*, in corrispondenza della voce di erogazione/fruizione in elenco. Le funzionalità di configurazione disponibili per ciascun sottoinsieme di azioni/risorse sono raggruppabili in:

- *Controllo Accessi*: per configurare i criteri di autenticazione, autorizzazione e gestione token delle richieste.
- *Rate Limiting*: per configurare i meccanismi di controllo del traffico a salvaguardia delle prestazioni.
- *Validazione*: per configurare i criteri di validazione dei messaggi in transito sul gateway.
- *Caching Risposta*: per configurare l'utilizzo della cache per i messaggi di risposta.

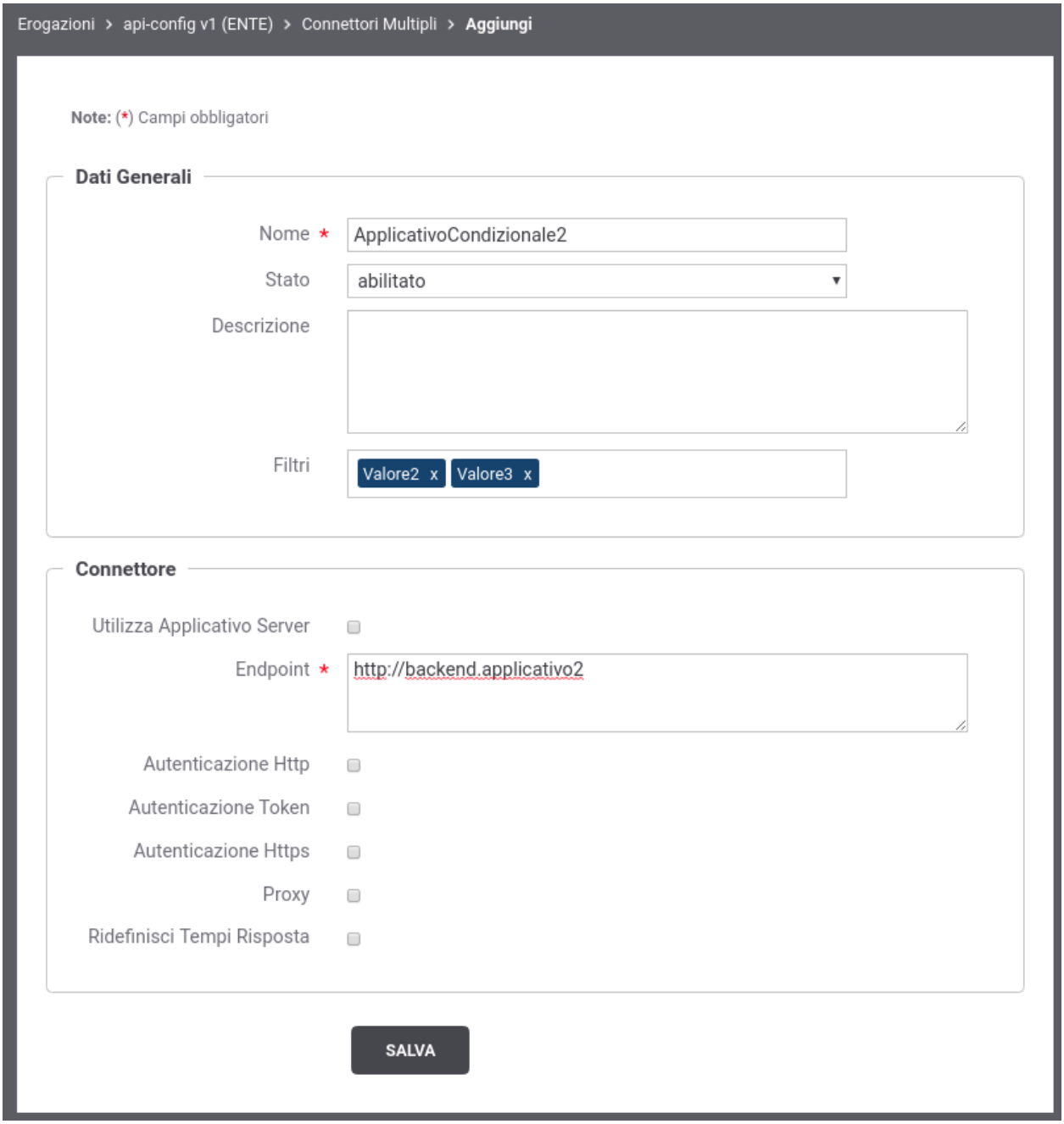


Fig. 2.51: Registrazione di un nuovo connettore per la consegna condizionale

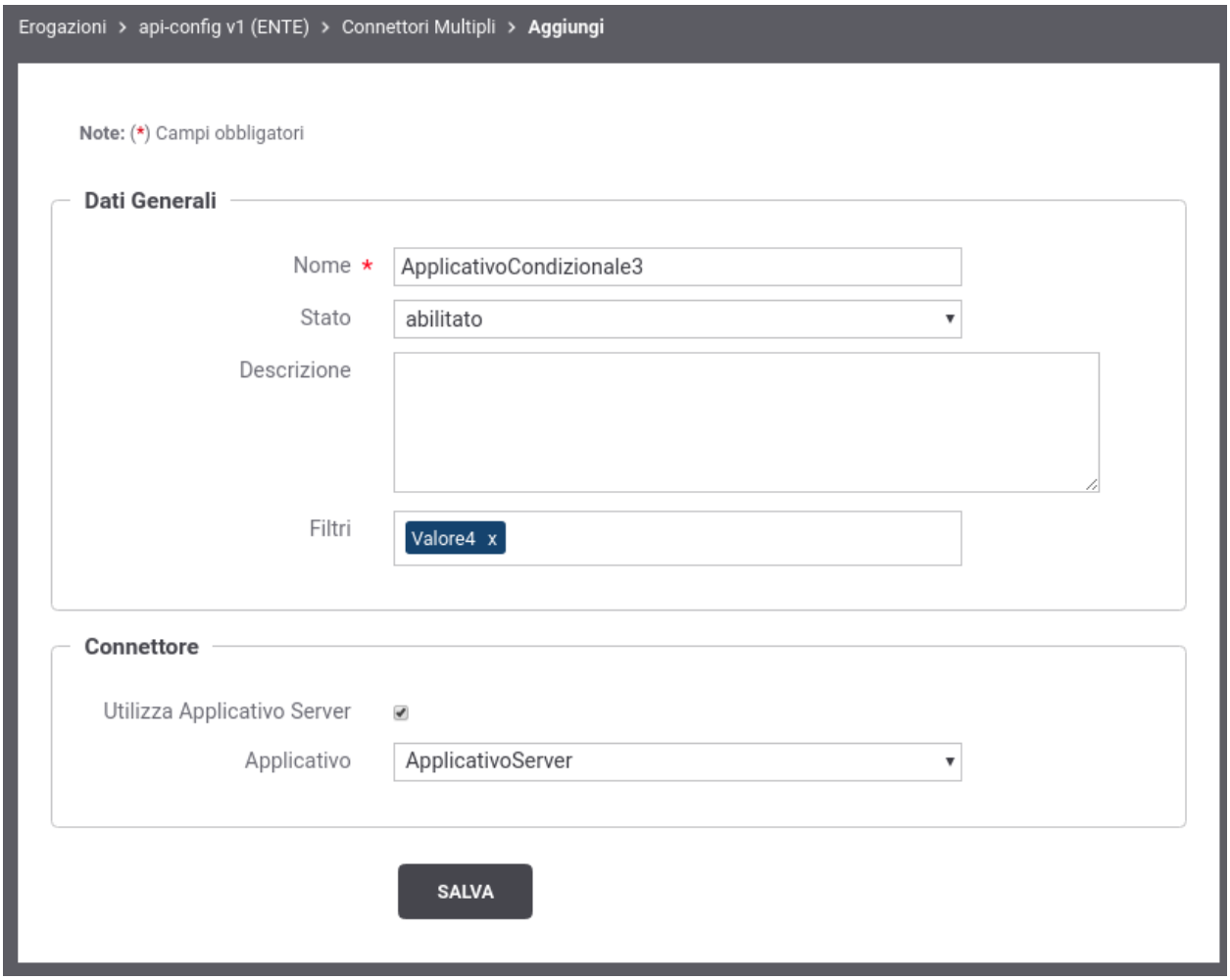


Fig. 2.52: Registrazione di un nuovo connettore, per la consegna condizionale, che riferisce un Applicativo Server

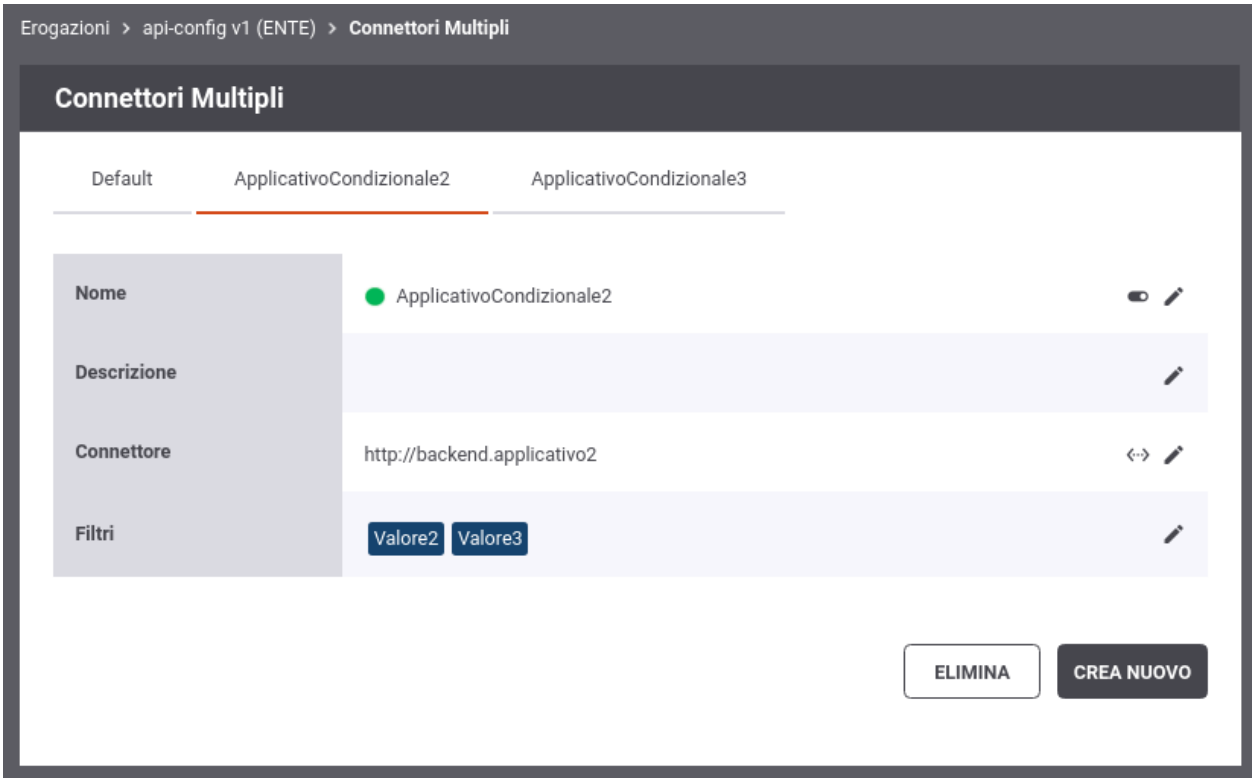


Fig. 2.53: Elenco dei connettori selezionabili per la consegna

- *Sicurezza Messaggio*: per configurare le misure di sicurezza applicate a livello del messaggio.
- *Tracciamento*: per configurare specifiche modalità di estrazione dati, dalle comunicazioni in transito, per l'arricchimento della traccia prodotta.
- *Trasformazioni*: per configurare le operazioni di trasformazione attivabili sui flussi in entrata ed uscita.
- *MTOM*: per configurare l'utilizzo del protocollo ottimizzato per l'invio di attachment tra nodi SOAP.
- *Registrazione Messaggi*: consente di ridefinire le politiche di archiviazione dei payload rispetto a quanto previsto dalla configurazione di default (vedi sezione *Tracciamento*).

Per creare un nuovo gruppo di configurazione, dopo aver seguito il collegamento *visualizza* relativo all'erogazione/fruizione selezionata, si preme il pulsante *Aggiungi*

Compilare il form di creazione della nuova configurazione (Fig. 2.54):

- *Azioni:* selezionare dall'elenco le azioni sulle quali si vuole abbia effetto la nuova configurazione.
- *Mode:* effettuare la scelta tra *Eredita Da* e *Nuova*. Scegliendo la prima opzione, verrà creata una configurazione clone di quella selezionata nell'elemento del form subito successivo (Configurazione). Scegliendo la seconda opzione, si procederà alla creazione di una nuova configurazione, specificando subito le informazioni di Controllo degli Accessi e Connettore.

Nota: Nota Dopo aver creato ulteriori configurazioni, si tenga presente che la configurazione di default verrà applicata alle sole azioni per le quali non è presente una regola di configurazione specifica.

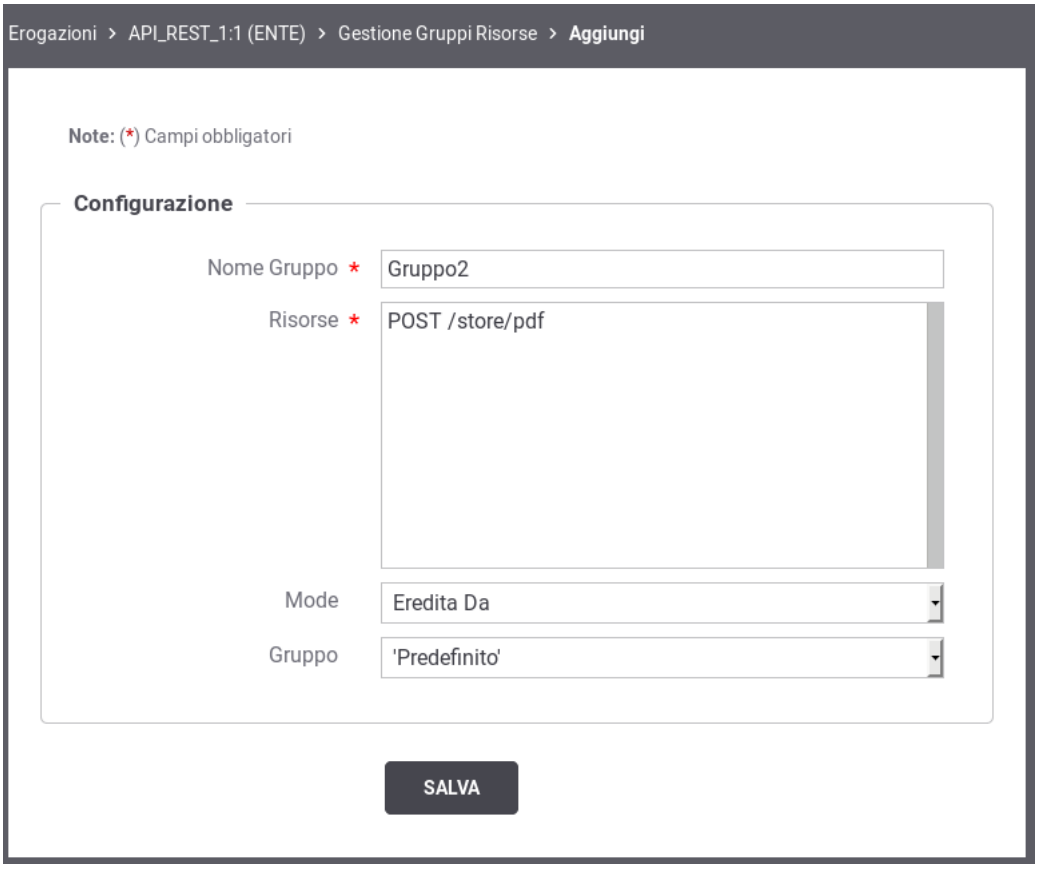


Fig. 2.54: Aggiunta di un gruppo di configurazioni

Nota: Nota È possibile disabilitare un'intera configurazione, senza la necessità di eliminarla, utilizzando il collegamento presente nella colonna «Abilitato» in corrispondenza dell'elemento di configurazione. Un successivo clic farà tornare la configurazione nello stato abilitato.

2.10 Controllo degli Accessi

Tramite questa funzionalità è possibile configurare i criteri di gestione token, autenticazione e autorizzazione delle richieste in ingresso sul gateway. Per aggiungere questa funzionalità si procede selezionando prima il collegamento, presente nella colonna «Configurazione», relativo all'erogazione/fruizione presente nell'elenco. Successivamente si utilizza il collegamento, presente nella colonna «Controllo Accessi», relativamente alla configurazione che si vuole modificare (Fig. 2.55).

Le sezioni seguenti descrivono le modalità per configurare gli aspetti che compongono il controllo degli accessi.

2.10.1 Autenticazione Token

Questa sezione consente di configurare il controllo degli accessi basato su Bearer Token OAuth2. Facendo transitare lo stato su «abilitato» compare l'elemento *Policy* (obbligatorio) per la selezione della policy di autenticazione token che si vuole applicare. In questa lista a discesa saranno visualizzate tutte le *Token Policy* di tipo *Validazione* che sono state registrate in precedenza. Per le istruzioni sulla registrazione delle Token Policy si faccia riferimento alla sezione *Token Policy*.

Una volta selezionata la policy compariranno sotto gli elementi per stabilire le specifiche azioni da abilitare rispetto al totale di quelle previste nella policy stessa (Fig. 2.56).

Supponendo che la policy copra tutti gli aspetti disponibili, le opzioni configurabili sono le seguenti:

- *Token Opzionale*: consente di non forzare i richiedenti al passaggio del token, che rimane quindi un'operazione opzionale.
- *Introspection*: consente di abilitare/disabilitare l'operazione di Token Introspection, al fine di validare il token ricevuto ed ottenere le metainformazioni associate (ad esempio scope e riferimento al possessore del token). Selezionando l'opzione *WarningOnly* è possibile non rendere bloccante l'evento di fallimento della validazione, ottenendo come unico effetto l'emissione di un messaggio diagnostico di segnalazione.
- *User Info*: consente di abilitare/disabilitare l'operazione UserInfo al fine di ottenere le informazioni di dettaglio dell'utente possessore del token. Selezionando l'opzione *WarningOnly* è possibile non rendere bloccante l'evento di fallimento della validazione, ottenendo come unico effetto l'emissione di un messaggio diagnostico di segnalazione.
- *Token Forward*: consente di abilitare/disabilitare l'operazione di inoltro, al servizio destinatario, del token ricevuto dal mittente.

Le azioni che sono state abilitate saranno effettuate in accordo a quanto configurato nella relativa Token Policy selezionata.

Nota: È disponibile la Token Policy *Google* preconfigurata in modo da utilizzare i servizi di elaborazione token esposti pubblicamente da Google e quindi:

- La Validazione JWT basata su *Google - ID Token* (https://www.googleapis.com/oauth2/v3/certs)

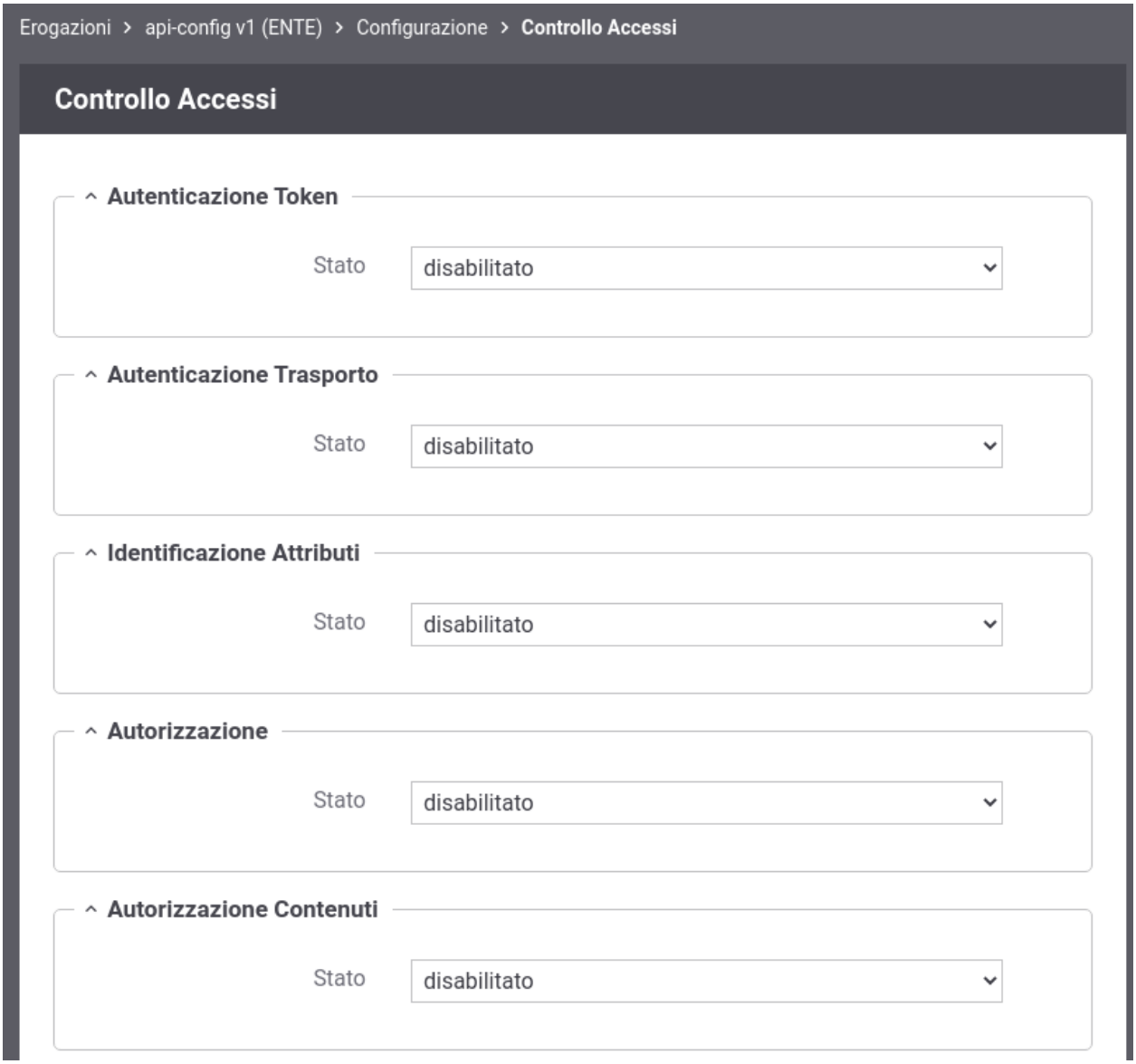


Fig. 2.55: Controllo degli Accessi

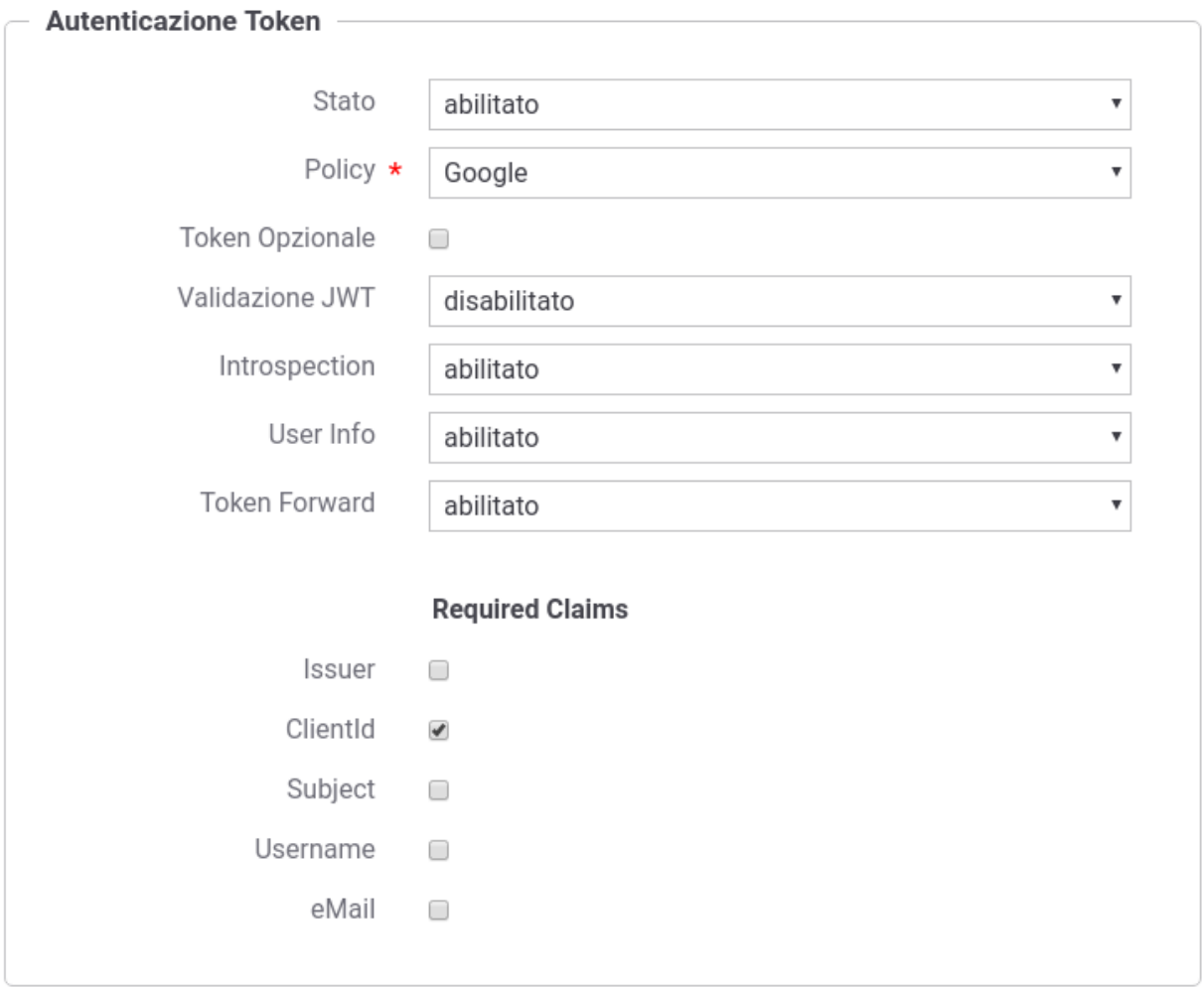


Fig. 2.56: Configurazione della gestione token

- Il servizio di token introspection basato su *Google - TokenInfo* (https://www.googleapis.com/oauth2/v3/tokeninfo)
- Il servizio di User Info basato su *Google - UserInfo* (https://www.googleapis.com/oauth2/v3/userinfo)

È possibile inoltre far verificare la presenza obbligatoria delle seguenti metainformazioni all'interno del token:

- Issuer
- ClientId
- Subject
- Username
- Email

2.10.2 Autenticazione Trasporto

In questa sezione è possibile configurare il meccanismo di autenticazione richiesto per l'accesso al servizio.

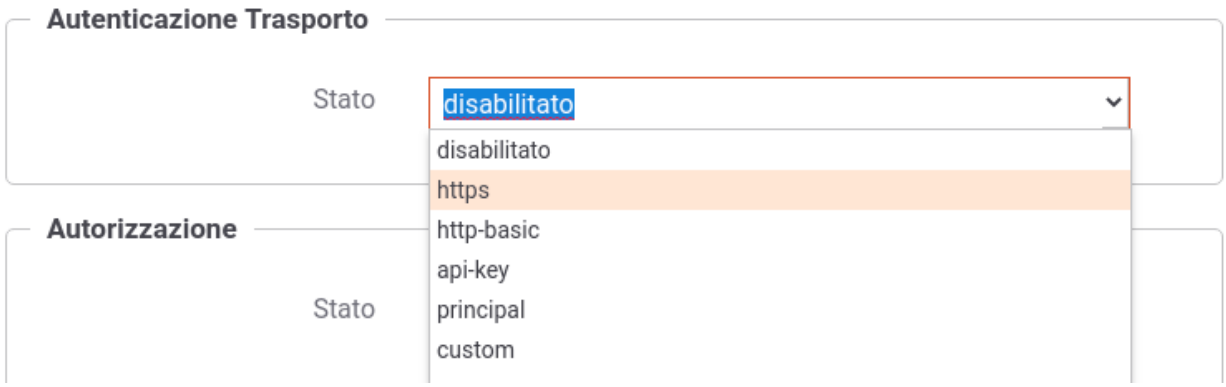


Fig. 2.57: Configurazione dell'autenticazione del servizio

Come mostrato in Fig. 2.57 la configurazione dell'autenticazione deve essere effettuata attraverso la selezione di un tipo di autenticazione tra quelli disponibili:

- disabilitato

Nessuna autenticazione.

- https

(Fig. 2.58) La richiesta deve possedere un certificato client X509. La presenza del certificato client nella richiesta è obbligatoria a meno che non sia abilitato il flag *Opzionale*. Per maggiori informazioni sulla configurazione necessaria affinchè il certificato client sia ricevuto dal gateway si faccia riferimento alla sezione install_ssl_server della "Guida di Installazione".

Se è presente un certificato client, il gateway cercherà di identificare un applicativo o un soggetto a cui è stato associato il certificato come credenziale di accesso (per ulteriori dettagli si rimanda alle sezioni *Creazione di un soggetto* e *Creazione di un applicativo*); l'identificazione non è obbligatoria ma nel caso avvenga con successo l'applicativo o il soggetto verrà registrato nei log e potrà essere utilizzato anche ai fini di autorizzazione puntuale e per ruoli (*Autorizzazione*).

- http-basic

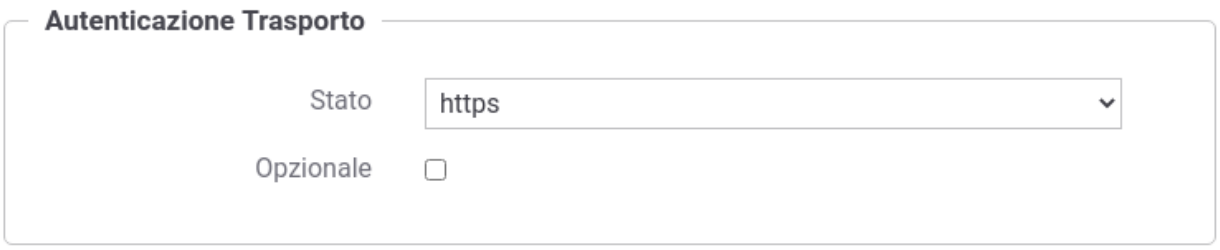


Fig. 2.58: Configurazione Autenticazione “https”

(Fig. 2.59) La richiesta deve possedere un header http «Authorization» che veicola credenziali Basic (username e password) come indicato nel rfc2617#section-2 (https://tools.ietf.org/html/rfc2617#section-2). La presenza dell’header «Authorization Basic» nella richiesta è obbligatoria a meno che non sia abilitato il flag *Opzionale*.

Abilitando l’ulteriore opzione *Forward Authorization* è possibile propagare all’endpoint di destinazione l’header http «Authorization» che altrimenti verrà consumata.

Le credenziali devono corrispondere ad un applicativo o un soggetto registrato nel gateway (per ulteriori dettagli si rimanda alle sezioni *Creazione di un soggetto* e *Creazione di un applicativo*).

Fig. 2.59: Configurazione Autenticazione “http-basic”

- api-key

(Fig. 2.60) La richiesta deve possedere una chiave di identificazione “Api Key” veicolata in un header http, un parametro della url o un cookie come indicato nella specifica “OAS3 API Keys” (https://swagger.io/docs/specification/authentication/api-keys/). È possibile abilitare anche la modalità “App ID” che prevede oltre all’ApiKey un identificatore dell’applicazione; modalità denominata “Multiple API Keys” nella specifica “OAS3 API Keys”. La presenza di una “Api Key”, e se attivata di una “App ID”, nella richiesta è obbligatoria a meno che non sia abilitato il flag *Opzionale*.

Abilitando le ulteriori opzioni *Forward* è possibile propagare all’endpoint di destinazione la chiave di identificazione ricevuta che altrimenti verrà consumata.

Le credenziali devono corrispondere ad un applicativo o un soggetto registrato nel gateway (per ulteriori dettagli si rimanda alle sezioni *Creazione di un soggetto* e *Creazione di un applicativo*).

La configurazione consente anche di indicare dove il gateway debba ricercare la chiave di accesso tra header http, parametro della url e cookie, permettendone anche di personalizzare i nomi che per default sono quelli indicati nella specifica OAS3 (Fig. 2.61).

- principal

(Fig. 2.63) La richiesta deve possedere il «principal» che identifica il chiamante. La modalità con cui il gateway può ottenere il principale deve essere scelta tra le seguenti opzioni:

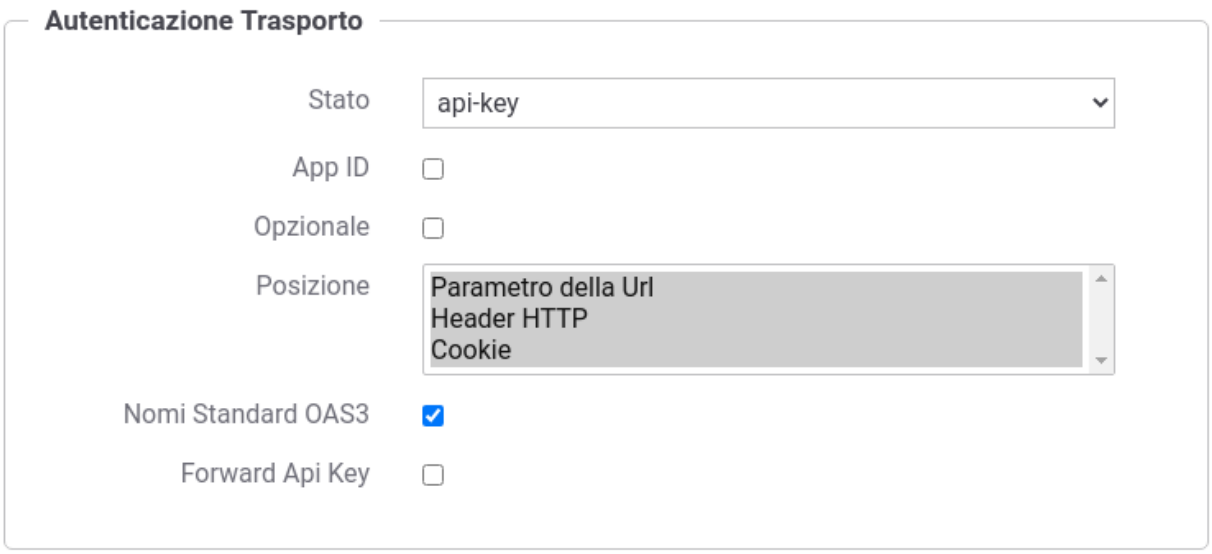


Fig. 2.60: Configurazione Autenticazione "api-key"

- *Container*: il principal viene fornito direttamente dal container sul quale è in esecuzione il gateway (per maggiori dettagli si faccia riferimento alla sezione *Autenticazione e Autorizzazione Principal (Security Constraint)*).
- *Header HTTP*: il principal viene estratto dallo specifico header http che viene indicato successivamente. È inoltre possibile attivare l'opzione *Forward Header* per far sì che il gateway propaghi il dato di autenticazione.
- *Parametro della Url*: il principal viene estratto da un parametro della query string il cui nome viene indicato successivamente. È inoltre possibile attivare l'opzione *Forward Parametro Url* per far sì che il gateway propaghi il dato di autenticazione.
- *Url di Invocazione*: il principal viene estratto direttamente dalla URL di invocazione tramite l'espressione regolare che viene fornita successivamente (l'espressione deve avere un match con l'intera url).
- *Client IP*: il principal utilizzato è l'indirizzo IP di provenienza.
- *X-Forwarded-For*: il principal viene estratto dall'header http utilizzato per il mantenimento dell'IP di origine nel caso di nodi intermedi (es. X-Forwarded-For).
- *Token*: opzione presente solamente se è stata attivata, al passo precedente, l'autenticazione del token. Il principal viene letto da uno dei claim presenti nel token.

Il flag *Opzionale* consente di non rendere bloccante il superamento dell'autenticazione nel caso la richiesta non possiede il principal atteso.

Se è presente un principal, il gateway cercherà di identificare un applicativo o un soggetto a cui è stato associato il principal come credenziale di accesso (per ulteriori dettagli si rimanda alle sezioni *Creazione di un soggetto* e *Creazione di un applicativo*); l'identificazione non è obbligatoria ma nel caso avvenga con successo l'applicativo o il soggetto verrà registrato nei log e potrà essere utilizzato anche ai fini di autorizzazione puntuale e per ruoli (*Autorizzazione*).

- plugin

Metodo di autenticazione fornito tramite personalizzazioni di GovWay.

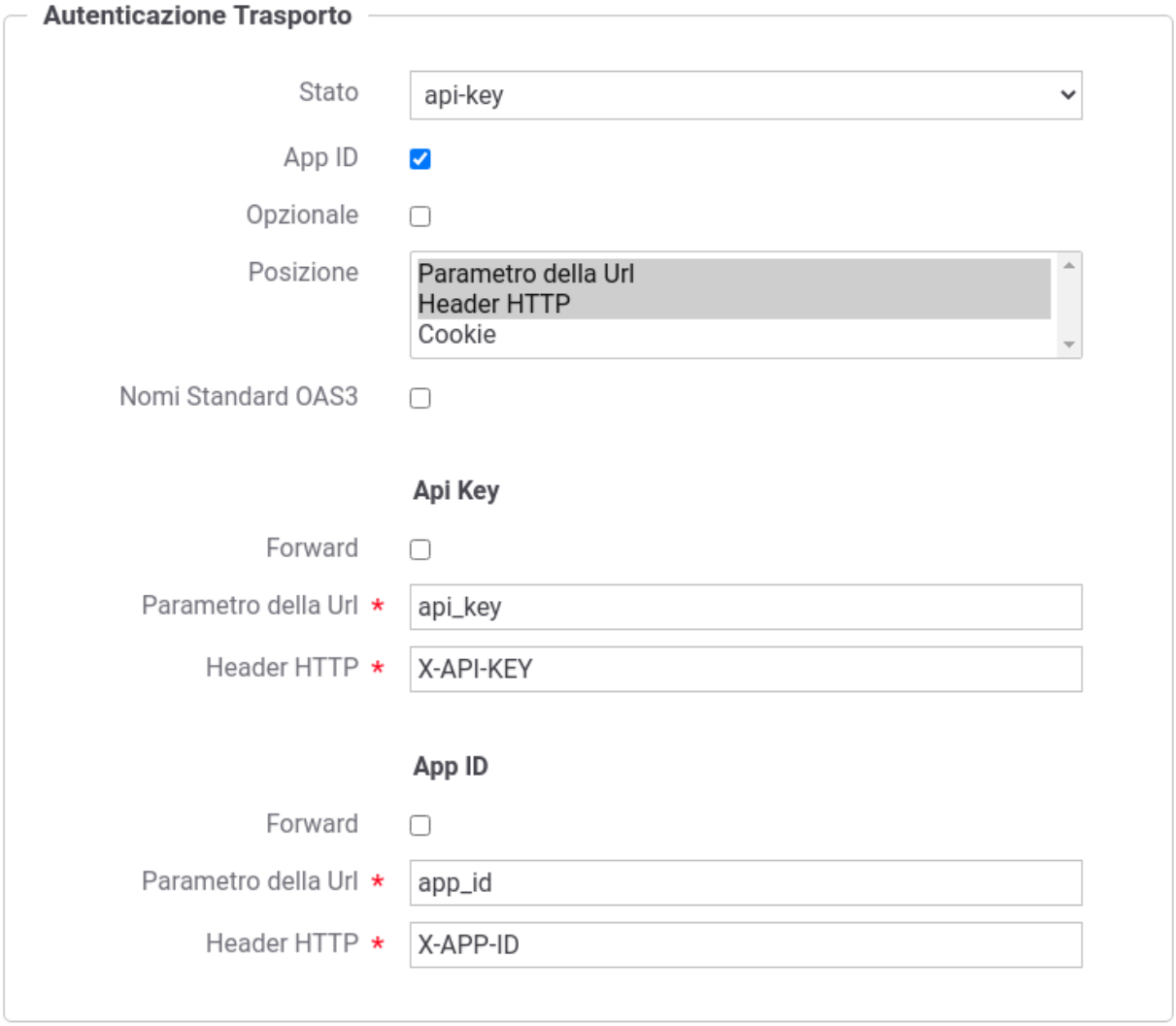


Fig. 2.61: Configurazione Autenticazione “api-key” con personalizzazione della posizione e dei nomi

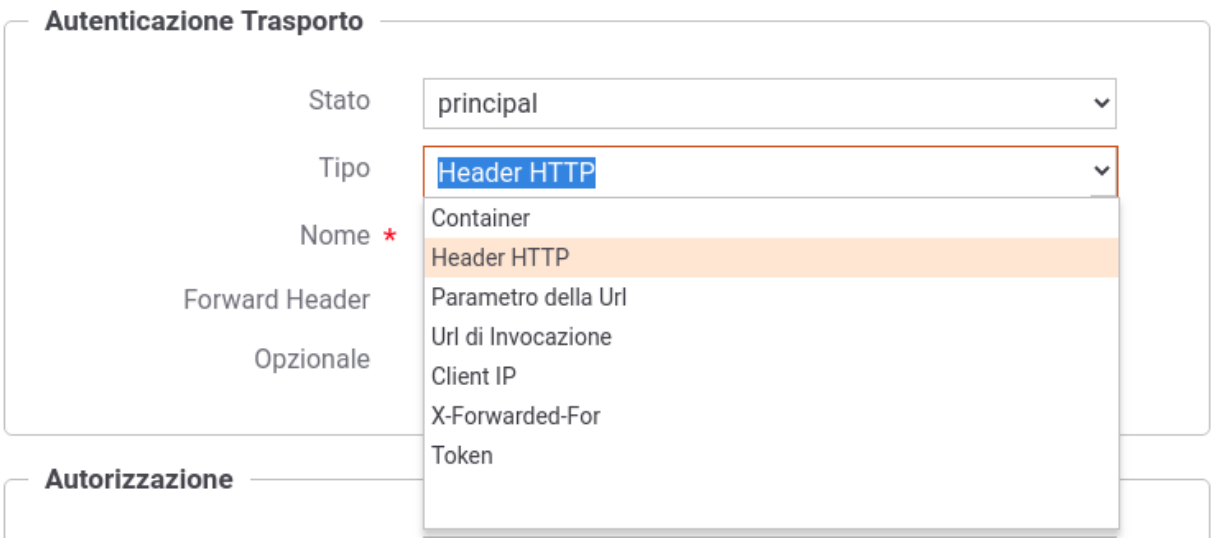


Fig. 2.62: Configurazione Tipo di Autenticazione “principal”

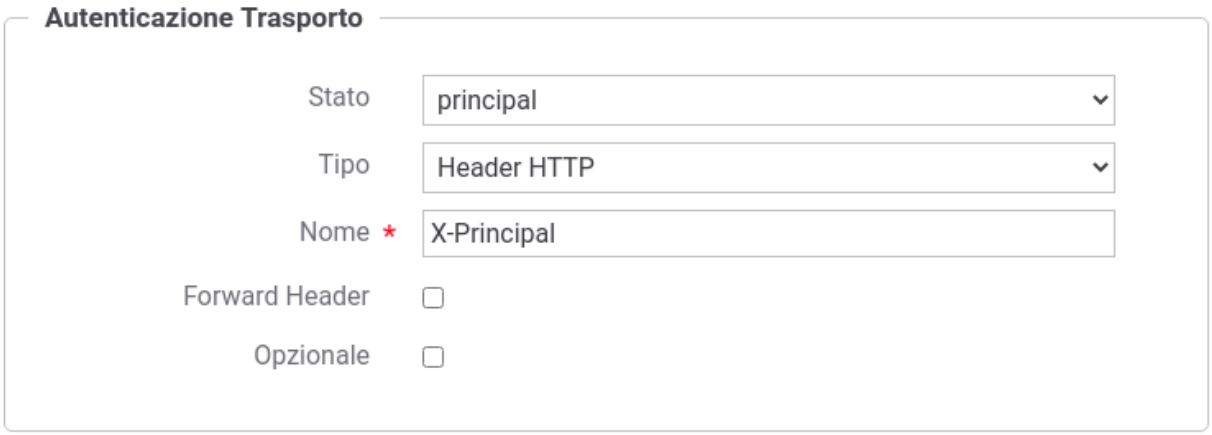


Fig. 2.63: Configurazione Autenticazione “principal”

2.10.3 Identificazione Attributi

Questa sezione consente di abilitare l'interrogazione di una o più *Attribute Authority* al fine di recuperare gli attributi qualificati del soggetto identificato su GovWay tramite i meccanismi di autenticazione precedentemente descritti nelle sezioni "*Autenticazione Token*" e "*Autenticazione Trasporto*".

Nota: La sezione viene visualizzata solamente se è stata registrata almeno una *Attribute Authority*.

Gli attributi recuperati saranno inseriti nel contesto della richiesta e potranno essere utilizzati per definire politiche di controllo degli accessi basate sugli attributi tramite i meccanismi di autorizzazione descritti nelle successive sezioni ("*XACML-Policy*", "*Autorizzazione per Token Claims*" e "*Autorizzazione Contenuti*".

Il form di configurazione appare come quello illustrato in Fig. 2.64.

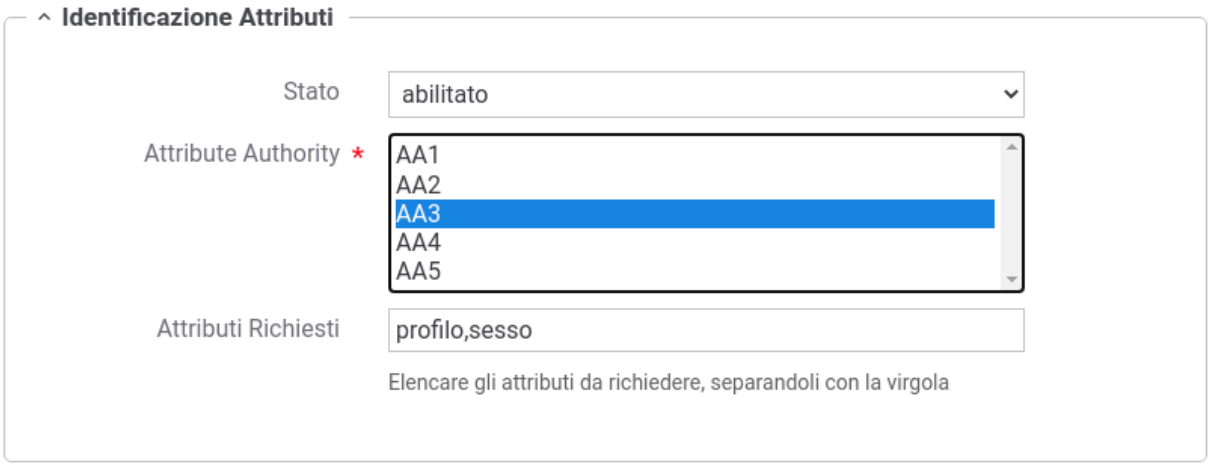


Fig. 2.64: Configurazione Identificazione Attributi tramite una singola AA

Una volta abilitata la funzionalità, se viene selezionata un'unica *Attribute Authority* sarà possibile indicare quali attributi qualificati debbano essere recuperati indicandoli nel campo "*Attributi Richiesti*", separandoli con la virgola, come riportato nell'esempio in Fig. 2.64.

Se invece vengono selezionate molteplici AA, gli attributi da richiedere devono essere indicati tramite una riga per ogni A.A. (nomeAA=listaAttributi) come in Fig. 2.65.

2.10.4 Autorizzazione

L'autorizzazione è un ulteriore meccanismo per il controllo degli accessi tramite il quale è possibile specificare con maggior dettaglio le richieste che sono in grado di essere accettate per l'accesso al servizio.

I meccanismi supportati, per specificare i criteri di autorizzazione, sono i seguenti:

- *Autorizzazione per Richiedente*: (Fig. 2.66) superato il processo di autenticazione, saranno accettate le sole richieste provenienti dai mittenti indicati singolarmente nella lista fornita con il criterio. Dopo aver abilitato questa opzione, ed aver confermato tramite il pulsante Invia, sarà possibile fornire la lista dei mittenti autorizzati ad accedere al servizio.

  I mittenti che possono essere indicati sono Soggetti (solo nel caso delle erogazioni) e Applicativi. Tali entità dovranno essere precedentemente registrate sulla govwayConsole seguendo le indicazioni fornite in sezione *Creazione di un soggetto* e *Creazione di un applicativo*.

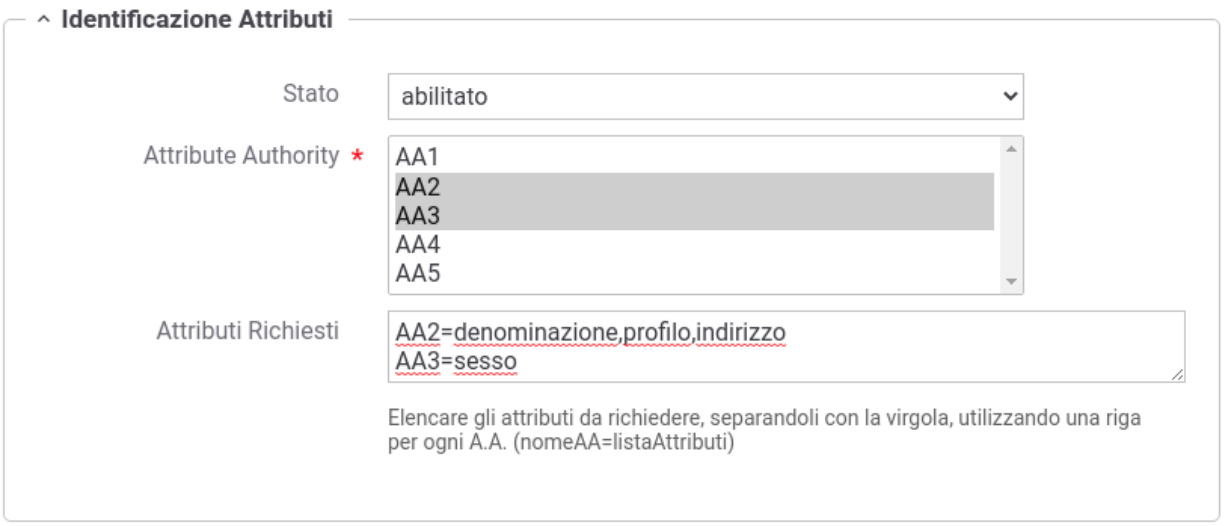


Fig. 2.65: Configurazione Identificazione Attributi tramite multiple AA

Nota: L'opzione di autorizzazione sui soggetti è disponibile solo se è stata attivata l'autenticazione.

Nota: L'opzione di autorizzazione sugli applicativi, nel caso di una erogazione, viene utilizzata per gestire l'accesso al servizio da parte di applicativi interni al dominio di GovWay.

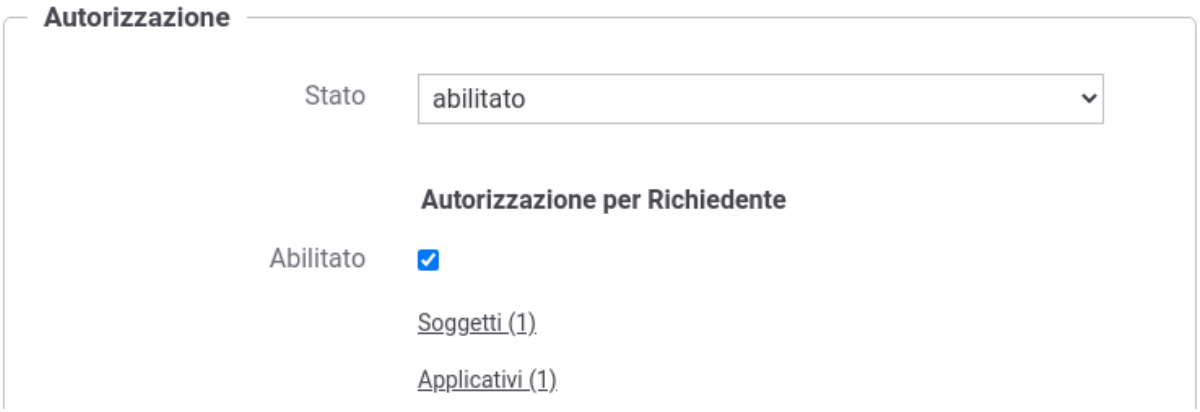


Fig. 2.66: Configurazione Autorizzazione per Richiedente

- *Autorizzazione per Ruoli*: (Fig. 2.67) consente di concedere l'autorizzazione per il servizio solo ai richiedenti in possesso di determinati ruoli nel proprio profilo. Dopo aver barrato questa opzione, ed aver confermato tramite il pulsante Invia, sarà possibile fornire una lista dei ruoli che devono essere posseduti dal chiamante per poter accedere al servizio. In particolare si dovrà anche specificare la *fonte* di provenienza dei ruoli, che può essere *esterna*, cioè proveniente dal sistema che ha autenticato il chiamante, oppure *registro*, cioè i ruoli che sono stati censiti nel registro di GovWay e assegnati al soggetto chiamante. Inoltre si deve scegliere l'opzione *Ruoli Richiesti* per indicare se, in presenza di più di un ruolo come criterio, il chiamante deve possedere «tutti» i ruoli indicati o «almeno uno».

  Per le indicazioni sul censimento dei ruoli fare riferimento alla sezione *Creazione di un ruolo*.

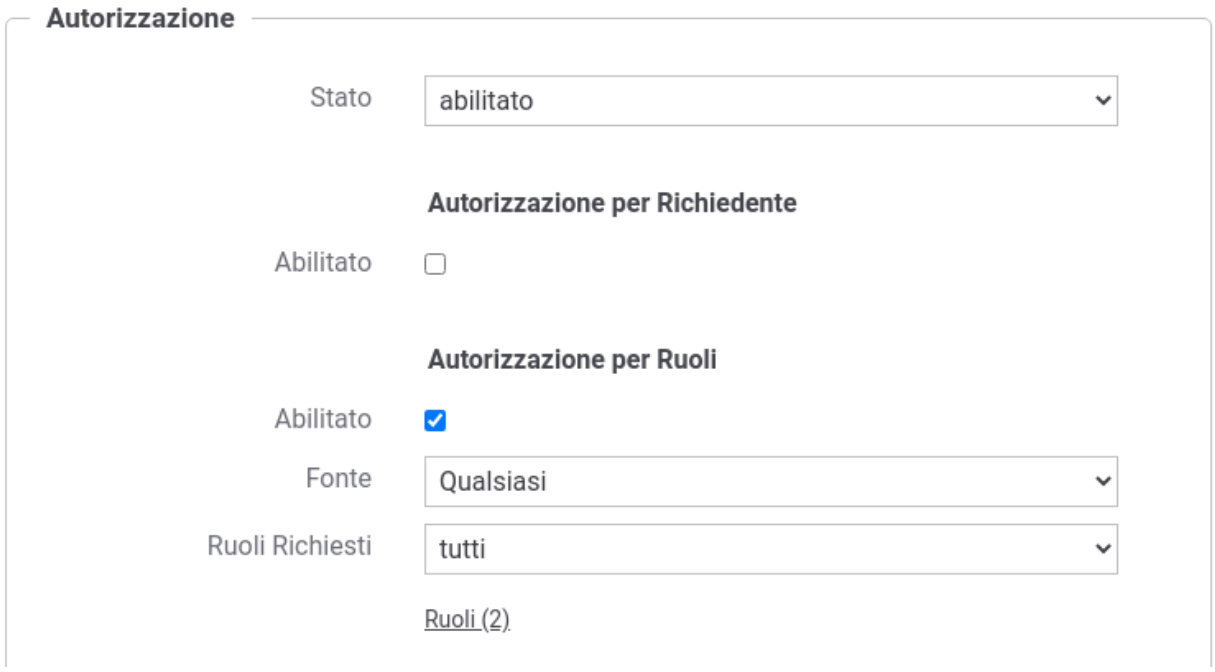


Fig. 2.67: Configurazione Autorizzazione per Ruoli

- *Autorizzazione per Scope*: (Fig. 2.68) criterio di autorizzazione che verifica la corrispondenza tra gli scope indicati e quelli estratti dal token presente nella richiesta ricevuta. Una volta attivata l'opzione si deve effettuare una scelta per l'elemento *Scope Richiesti*, tra i valori «tutti» (tutti gli scope indicati devono essere presenti nel token per superare l'autorizzazione) e «almeno uno» (è richiesta la presenza di almeno uno scope tra quelli indicati nella policy di autorizzazione). Dopo aver confermato la scelta con il pulsante «Invia» verrà richiesto di inserire gli scope tra quelli già censiti ed abilitati per l'uso nei contesti di erogazione (o qualsiasi contesto).

  Per le indicazioni sul censimento degli scope fare riferimento alla sezione *Scope*.

  Nota: L'opzione di autorizzazione basata sugli scope è disponibile solo se è stata preventivamente attivata la Gestione Token e selezionata la relativa policy.

- *Autorizzazione per Token Claims*: (Fig. 2.69) Se è stata abilitata la gestione del token si ha la possibilità di autorizzare le richieste inserendo i valori ammessi per i claims contenuti nel token. La configurazione viene effettuata inserendo nel campo di testo ciascun claim in una riga, facendo seguire dopo l'uguale i valori ammessi separati da virgola.

  Per le indicazioni di dettaglio sui possibili controlli effettuabili su ogni claim si faccia riferimento alla sezione *Autorizzazione per Token Claims*.

  Nota: L'opzione di autorizzazione basata sui token è disponibile solo se è stata preventivamente attivata la Gestione Token e selezionata la relativa policy.

- *XACML-Policy*: (Fig. 2.70) È possibile basare il meccanismo di autorizzazione sulla valutazione di una policy xacml selezionando la relativa opzione sulla lista «Stato».

  Per le indicazioni di dettaglio sulla configurazione delle xacml-Policy si faccia riferimento alla sezione *XACML-Policy*.

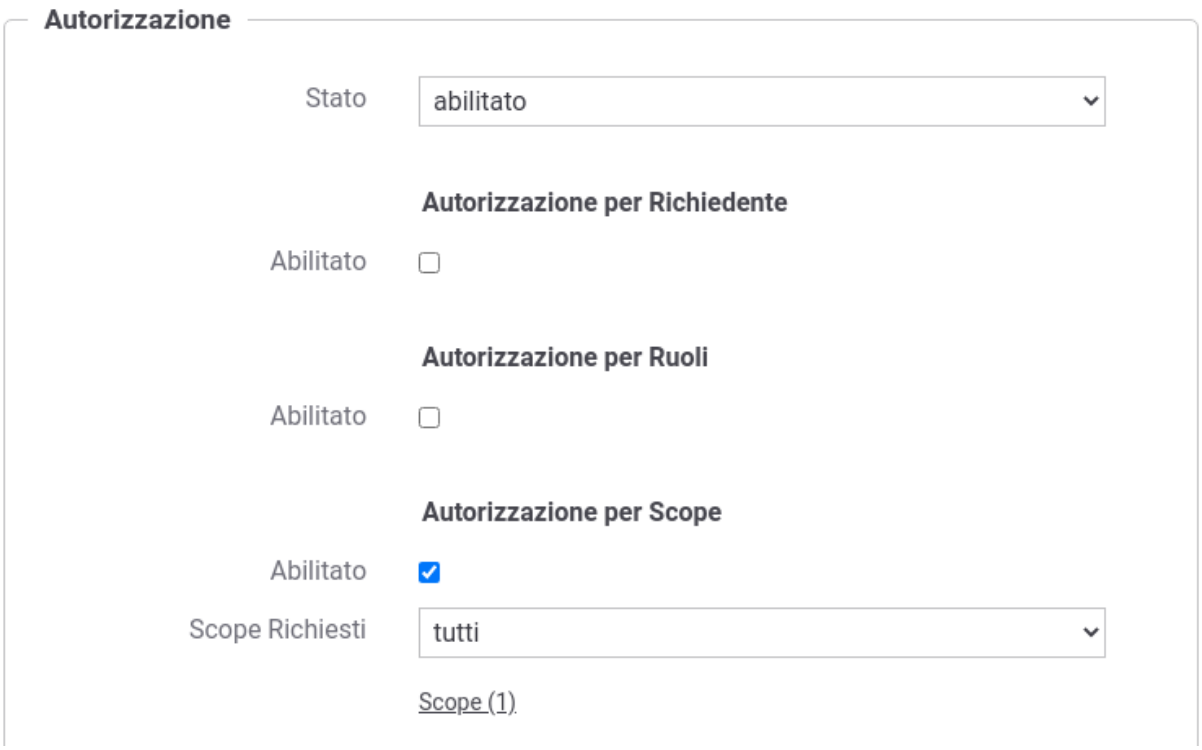


Fig. 2.68: Configurazione Autorizzazione per Scope

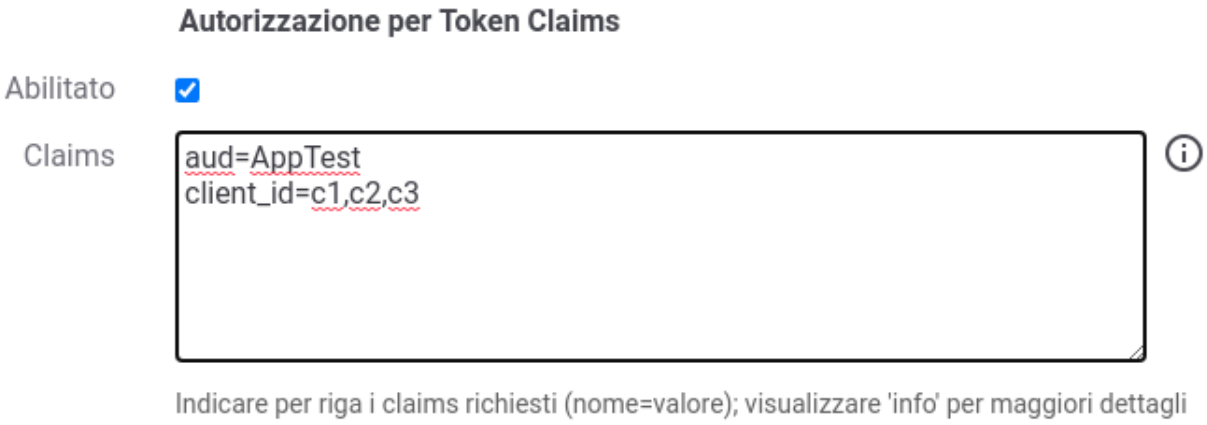


Fig. 2.69: Configurazione Autorizzazione per Token Claims

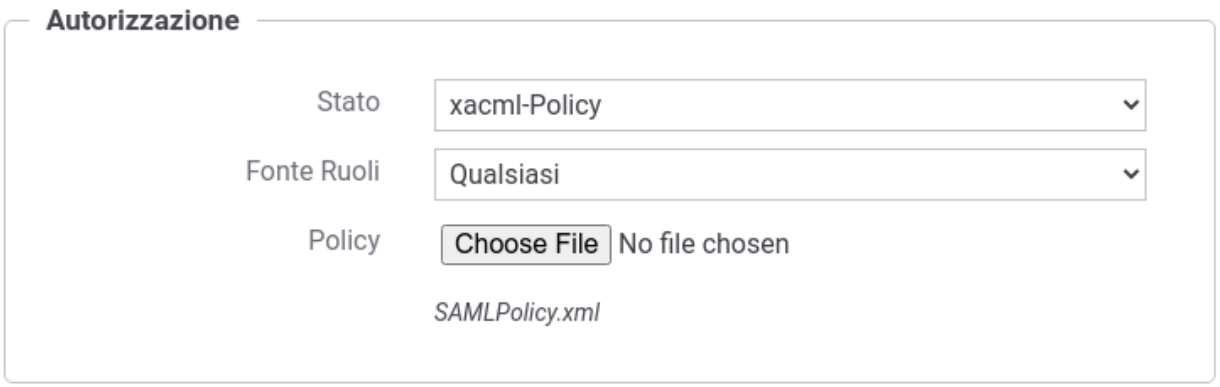


Fig. 2.70: Configurazione Autorizzazione XACML-Policy

- *Custom*: Sulla lista «Stato», è possibile selezionare questo metodo di autorizzazione eventualmente fornito tramite estensione di GovWay.

2.10.5 Autorizzazione Contenuti

L'autorizzazione dei contenuti è un ulteriore meccanismo per il controllo degli accessi tramite il quale è possibile specificare regole di autorizzazione che verificano aspetti della richiesta quali ad esempio gli header http, l'url di invocazione, parti del messaggio etc.

Una volta abilitata l'autorizzazione per contenuto si possono configuare una serie di controlli di autorizzazione nella forma (risorsa=valore).

Una risorsa identifica un header, una parte dell'url o del messaggio, un claim del token o un principal etc. Per identificare una risorsa sono utilizzabili le seguenti espressioni dinamiche:

- ${header:NAME}: valore presente nell'header http che possiede il nome "NAME"
- ${query:NAME}: valore associato al parametro della url con nome "NAME"
- ${urlRegExp:EXPR}: espressione regolare applicata sulla url di invocazione (l'espressione deve avere un match con l'intera url)
- ${xPath:EXPR}: espressione XPath applicata su un messaggio XML
- ${jsonPath:EXPR}: espressione JSONPath applicata su un messaggio JSON
- ${tokenInfo:FIELD}: permette di accedere ai claim di un token; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.pdd.core.token.InformazioniToken" (es. per ottenere il valore del claim "sub" usare ${tokenInfo:sub})
- ${aa:FIELD} : permette di accedere agli attributi recuperati tramite Attribute Authority; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.pdd.core.token.attribute_authority.InformazioniAttributi" (es. per ottenere il valore dell'attributo "attr1" usare ${aa:attributes[attr1]}, se configurata solamente 1 A.A., altrimenti usare ${aa:attributes[nomeAttributeAuthority][attr1]})
- ${transportContext:FIELD}: permette di accedere ai dati della richiesta http; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.utils.transport.http.HttpServletTransportRequestContext" (es. per il principal usare ${transportContext:credential.principal})
- ${config:NAME}: valore della proprietà configurata sull'API che possiede il nome "NAME"

- ${clientApplicationConfig:NAME}: valore della proprietà configurata nell'applicativo fruitore che possiede il nome "NAME"
- ${clientOrganizationConfig:NAME}: valore della proprietà configurata nel soggetto fruitore che possiede il nome "NAME"
- ${providerOrganizationConfig:NAME}: valore della proprietà configurata nel soggetto erogatore che possiede il nome "NAME"
- ${system:NAME}: valore associato alla proprietà di sistema, indicata nella configurazione generale, con nome "NAME"
- ${env:NAME}: valore associato alla variabile di sistema con nome "NAME"
- ${java:NAME}: valore associato alla variabile java con nome "NAME"

Ogni valore atteso per una risorsa può essere fornito in una delle seguenti modalità:

- ${anyValue} : indica qualsiasi valore non nullo
- ${undefined} : la risorsa indicata non deve esistere o non deve essere valorizzata
- ${regExpMatch:EXPR} : la regola è soddisfatta se il valore della risorsa ha un match completo rispetto all'espressione regolare EXPR indicata
- ${regExpFind:EXPR} : simile alla precedente regola, il match dell'espressione regolare può avvenire anche su una sottostringa del valore della risorsa
- valore : indica esattamente il valore (case sensitive) che deve possedere la risorsa; il valore può essere definito come costante o contenere parti dinamiche risolte a runtime dal Gateway nella forma descritta precedentemente per le risorse
- valore1,..,valoreN : è possibile elencare differenti valori ammissibili; come per la precedente opzione il valore può contenere parti dinamiche

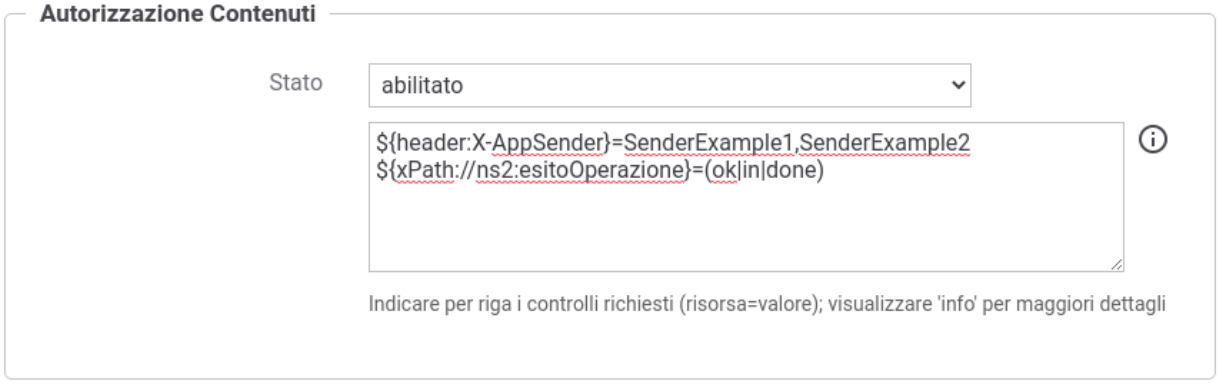


Fig. 2.71: Configurazione Autorizzazione Contenuti

Di seguito alcuni esempi:

- ${header:X-Prova}=test : viene verificato che l'header "X-Prova" possieda il valore "test"
- ${header:X-Prova}=test,test2,test3 : viene verificato che l'header "X-Prova" possieda il valore "test" o "test2" o "test3"
- ${transportContext:credential.principal}=${header:X-SSO} : viene verificato che l'identità principal del chiamante corrisponda al valore fornito nell'header "X-SSO"

- ${transportContext:credential.principal}=prefix${header:X-SSO}suffix : simile alla precedente regola, dove l'identità principal viene controntata con il valore presente nell'header concatenato da un prefisso e da un suffisso statico.
- ${xPath:EXPR}=${regExpMatch:[0-9]} : viene estratto il contenuto dalla richiesta xml tramite l'espressione XPath EXPR e verificato che sia corrispondente ad una cifra decimale attraverso l'espressione regolare "[0-9]"
- ${jsonPath:EXPR}=${transportContext:credential.principal} : viene estratto il contenuto dalla richiesta json tramite l'espressione jsonPath EXPR e verificato che sia uguale all'identità principal del chiamante
- ${context:CLIENT_IP_REMOTE_ADDRESS}=10.114.44.3,10.114.44.4,10.114.44.5 : viene verificato che l'indirizzo ip del client sia tra gli indirizzi ip elencati.
- ${context:CLIENT_IP_TRANSPORT_ADDRESS}=${regExpMatch:10.114.44..*|10.114.43..*} : viene verificato che l'indirizzo ip del client sia nella sottorete 10.114.44.0/255 o 10.114.43.0/255; l'indirizzo ip viene estratto dagli header http utilizzati per il mantenimento dell'IP di origine nel caso di nodi intermedi (es. X-Forwarded-For).

2.10.6 Creazione di un soggetto

Affinché possano essere utilizzate le funzionalità di autenticazione ed autorizzazione, associate alle erogazioni, è necessario che vengano censiti i soggetti fruitori che inviano le richieste di servizio. La registrazione di un soggetto consente di assegnargli delle credenziali che lo identificano ed eventuali ruoli provenienti dalla fonte «Registro».

Per creare il soggetto posizionarsi nella sezione *Registro* > *Soggetti*, quindi premere il pulsante *Aggiungi*. Compilare il form come segue (Fig. 2.72):

- *Profilo Interoperabilità*: La scelta del profilo di interoperabilità sarà richiesta solo nel caso in cui non sia stata effettuata la relativa scelta dal menu in testata.
- *Nome*: Il nome del soggetto. È necessario che il nome indicato risulti univoco rispetto ai nomi già presenti per la modalità operativa selezionata (in questo caso API Gateway).
- *Tipologia*: Indicare se si tratta di un soggetto esclusivamente erogatore, esclusivamente fruitore o con entrambi i ruoli.
- *Descrizione*: Un testo di descrizione per il soggetto.
- *Modalità di Accesso*: Sezione presente solo nel caso in cui il soggetto ricopra il ruolo di fruitore. Tramite il campo *Tipo* si seleziona il tipo di credenziali richieste per l'autenticazione del soggetto. In base alla scelta effettuata saranno mostrati i campi per consentire l'inserimento delle credenziali richieste. Per i dettagli sulla configurazione della modalità di accesso si faccia riferimento alla sezione *Modalità di Accesso*.

Dopo aver creato il soggetto è opzionalmente possibile assegnargli dei ruoli, tra quelli che sono presenti nel registro e contrassegnati come *fonte registro*. Per associare ruoli ad un soggetto seguire il collegamento presente nella colonna *Ruoli*, in corrispondenza del soggetto scelto. Premere quindi il pulsante *Aggiungi*. Nel form che si apre (Fig. 2.73) è presente una lista dalla quale è possibile selezionare un ruolo alla volta, che viene aggiunto confermando con il tasto *Invia*.

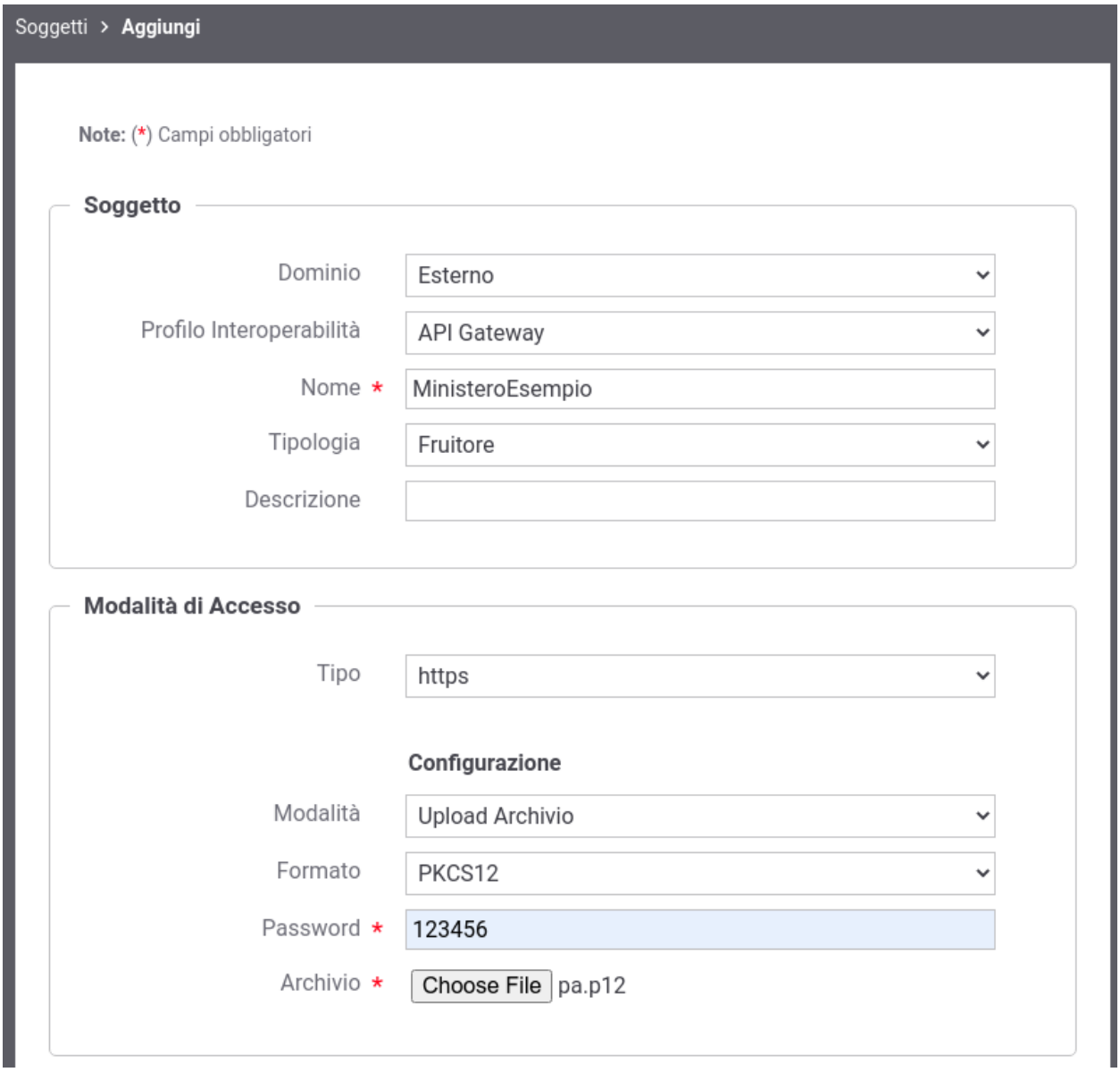


Fig. 2.72: Creazione di un soggetto

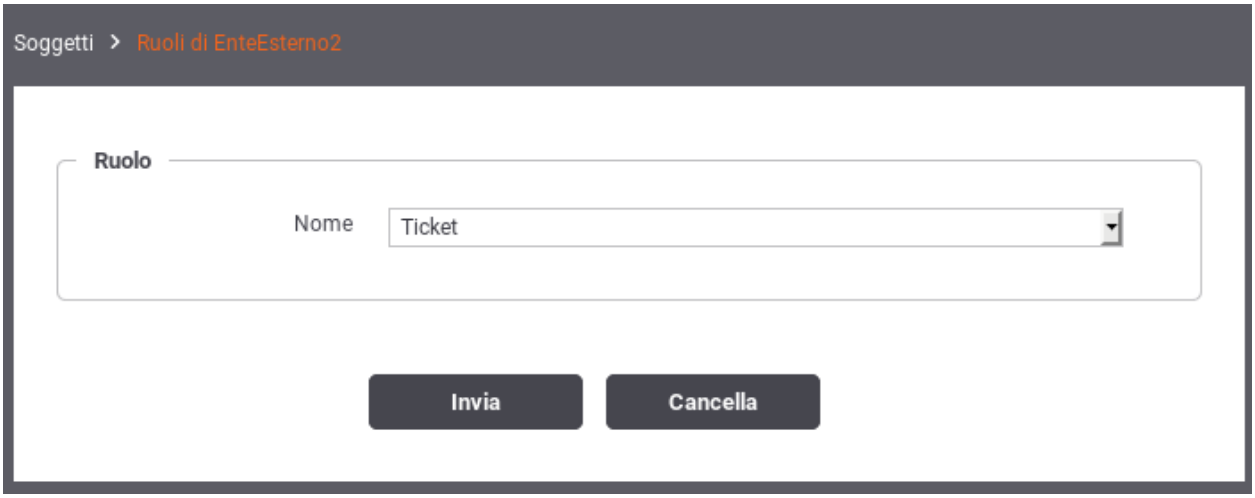


Fig. 2.73: Assegnazione di ruoli ad un soggetto

2.10.7 Creazione di un applicativo

Affinché possano essere utilizzate le funzionalità di autenticazione ed autorizzazione, associate alle fruizioni, è necessario che vengano censiti gli applicativi client, interni al dominio, che inviano le richieste di servizio. La registrazione di un applicativo, di tipo client, consente di assegnargli delle credenziali che lo identificano ed eventuali ruoli provenienti dalla fonte «Registro».

Per registrare l'applicativo posizionarsi nella sezione *Registro* > *Applicativi*, quindi premere il pulsante *Aggiungi*. Compilare il form come segue (Fig. 2.74):

- *Profilo Interoperabilità*: Opzione visibile solo nel caso in cui non sia stata effettuata la relativa scelta sul menu della testata.
- *Nome*: Assegnare un nome all'applicativo. È necessario che il nome indicato risulti univoco rispetto ai nomi già presenti per la modalità operativa selezionata (in questo caso API Gateway).
- *Tipo*: Utilizzare il tipo "Client" per censire un'applicativo allo scopo di identificarlo ed autorizzarlo durante l'invocazione di erogazioni o fruizioni di API.
- *Modalità di Accesso*: Tramite il campo *Tipo* si seleziona il tipo di credenziali richieste per l'autenticazione dell'applicativo. In base alla scelta effettuata saranno mostrati i campi per consentire l'inserimento delle credenziali richieste. Per i dettagli sulla configurazione della modalità di accesso si faccia riferimento alla sezione *Modalità di Accesso*.

Dopo aver creato l'applicativo è opzionalmente possibile assegnargli dei ruoli, tra quelli che sono presenti nel registro e contrassegnati come *fonte registro*. Per associare ruoli ad un applicativo seguire il collegamento presente nella colonna *Ruoli*, in corrispondenza dell'applicativo scelto. Premere quindi il pulsante *Aggiungi*. Nel form che si apre (Fig. 2.75) è presente una lista dalla quale è possibile selezionare un ruolo alla volta, che viene aggiunto confermando con il tasto *Invia*.

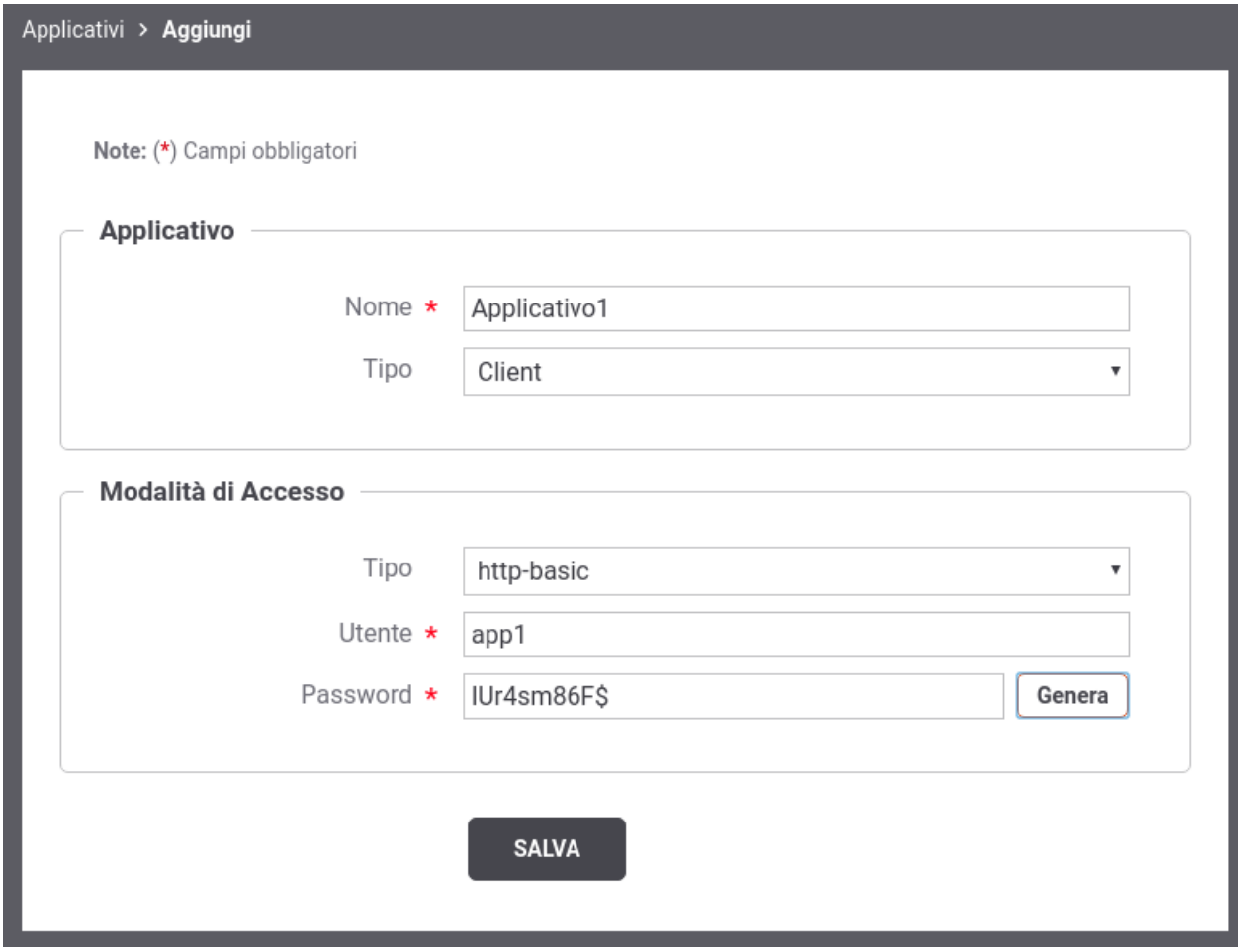


Fig. 2.74: Creazione di un applicativo

Fig. 2.75: Assegnazione di ruoli ad un applicativo

2.10.8 Modalità di Accesso

Agli applicativi ed ai soggetti registrati nel gateway (come indicato nelle sezioni *Creazione di un soggetto* e *Creazione di un applicativo*) devono essere assegnate delle credenziali in modo che il gateway possa effettuare:

- *autenticazione*: nel caso in cui il controllo degli accessi sia stato configurato con autenticazione "http-basic" o "api-key" (*Autenticazione Trasporto*)
- *identificazione*: l'identificazione non è obbligatoria per le autenticazioni differenti da "http-basic" e "api-key", ma nel caso avvenga con successo l'applicativo o il soggetto verrà registrato nei log e potrà essere ricercato tramite gli strumenti di monitoraggio
- *autorizzazione*: se un applicativo o un soggetto viene identificato, può essere autorizzato puntualmente nel controllo degli accessi tramite l'autorizzazione per richiedente (*Autorizzazione*).

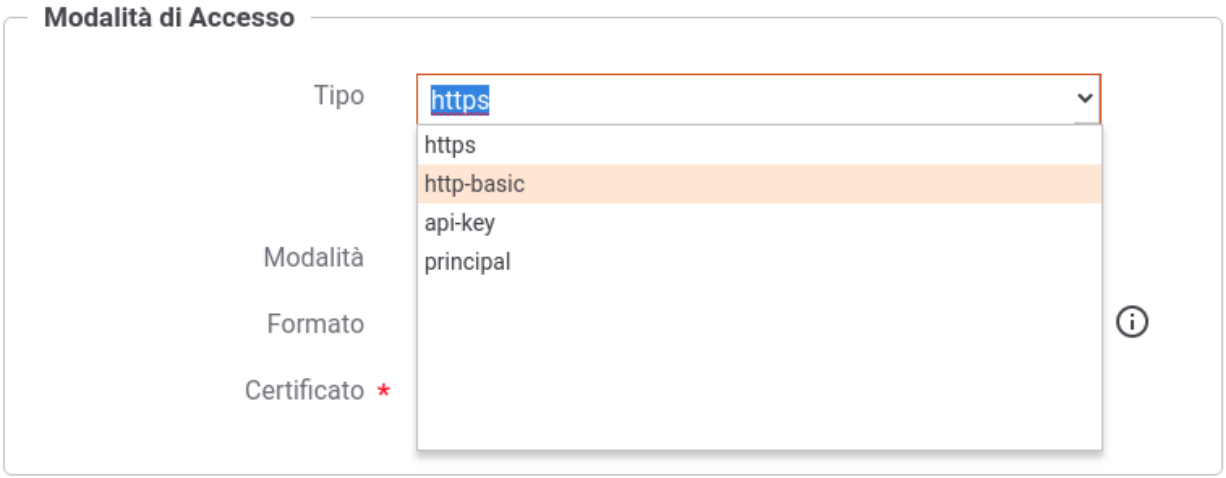


Fig. 2.76: Configurazione della Modalità di Accesso

Come mostrato in Fig. 2.76 l'assegnazione delle credenziali deve essere effettuata attraverso la selezione di un tipo di autenticazione tra quelli disponibili:

- *https*: richiede la registrazione di un certificato client X509
- *http-basic*: deve essere definito un username univoco e deve essere generata una password
- *apikey*: richiede la generazione di una chiave di identificazione univoca
- *principal*: deve essere assegnato un identificatore univoco

Credenziali "https"

Agli applicativi ed ai soggetti registrati nel gateway, identificabili con credenziali "https", deve essere associato un certificato client X509.

Per la configurazione si procede selezionando dall'elemento *Modalità* una tra le seguenti opzioni:

- **Upload Archivio** (Fig. 2.77): con questa modalità di configurazione si procede con il caricamento del certificato che sarà utilizzato per l'autenticazione. È necessario indicare:
  - *Formato*: il formato del certificato fornito specificando tra le seguenti opzioni supportate:
    - *CER*: il certificato da caricare è in formato *DER* o *PEM*.
    - *JKS*: il certificato da caricare è contenuto in un keystore JKS.

    * *PKCS12*: il certificato da caricare è contenuto in un keystore PKCS12.
  - *Password*: campo visibile nel caso in cui il certificato da caricare è contenuto in un keystore JKS o PCKS12. Rappresenta la password per l'accesso al keystore.
  - *Archivio*: selezionare dal proprio filesystem il file che contiene il certificato.
  - *Alias*: nel caso in cui il keystore contenga più di un certificato (frequente in formati JKS), questa lista consente di selezionare l'alias che riferisce l'elemento corretto.

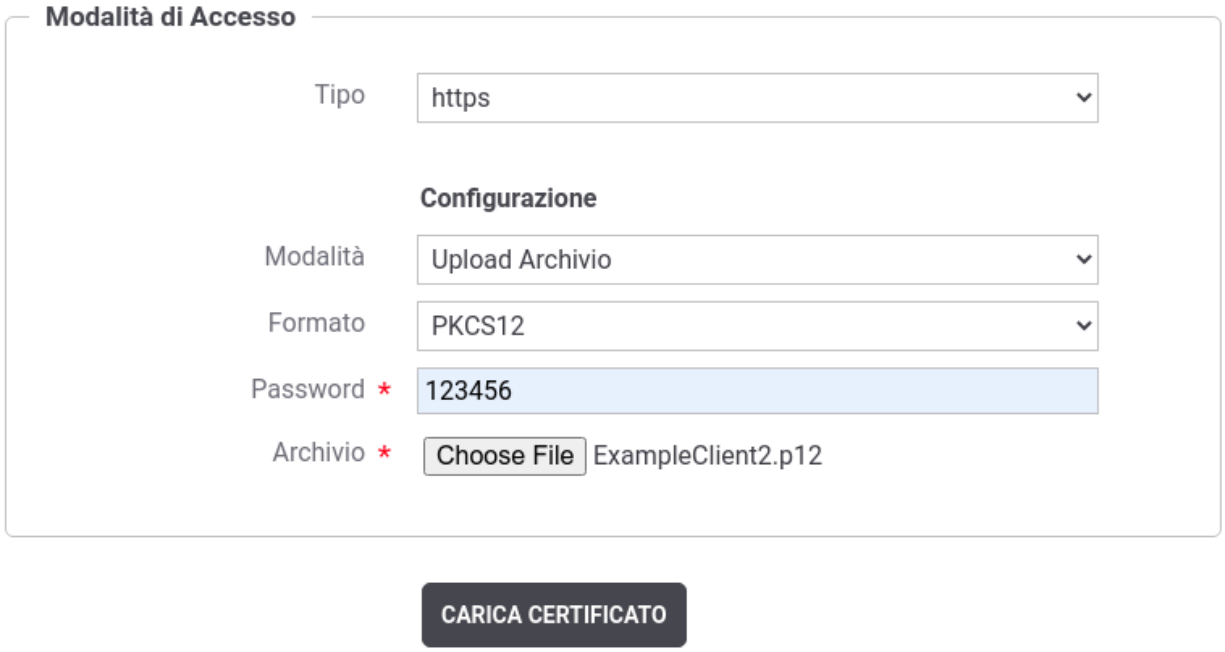


Fig. 2.77: Credenziali di tipo HTTPS (upload archivio 1/2)

Una volta caricato l'archivio verranno mostrati a video i dettagli del certficato selezionato (Fig. 2.78), al fine di poterli verificare prima di confermare l'inserimento. Il certficato caricato verrà confrontato con il certificato fornito durante l'autenticazione se la voce *Verifica* è abilitata, altrimenti verranno controllati solamente che i DN del Subject e dell'Issuer siano identici. Un confronto fallito causeranno il fallimento dell'autenticazione.

Dopo aver creato un applicativo o un soggetto con associato un certificato, visualizzandone i dati è possibile effettuare il download del certificato o aggiungerne di ulteriori.

La funzionalità di aggiunta di un certificato può essere utilizzata per gestire preventivamente la scadenza di un certificato caricando anche la versione aggiornata in modo da poter essere in grado di autenticare l'applicativo non appena inizia ad utilizzare il nuovo certificato. Sia in fase di aggiunta che successivamente sarà possibile promuovere a "principale" la versione aggiornata del certificato ed eliminare successivamente la versione scaduta.

- **Configurazione Manuale** (Fig. 2.81): con questa modalità di configurazione si procede con l'inserimento dei seguenti dati:
  - *Self Signed*: opzione per indicare se il cerfificato è self-signed oppure rilasciato da una CA.
  - *Subject*: il subject del certificato.
  - *Issuer*: l'issuer del certificato, nel caso in cui non sia self-signed.

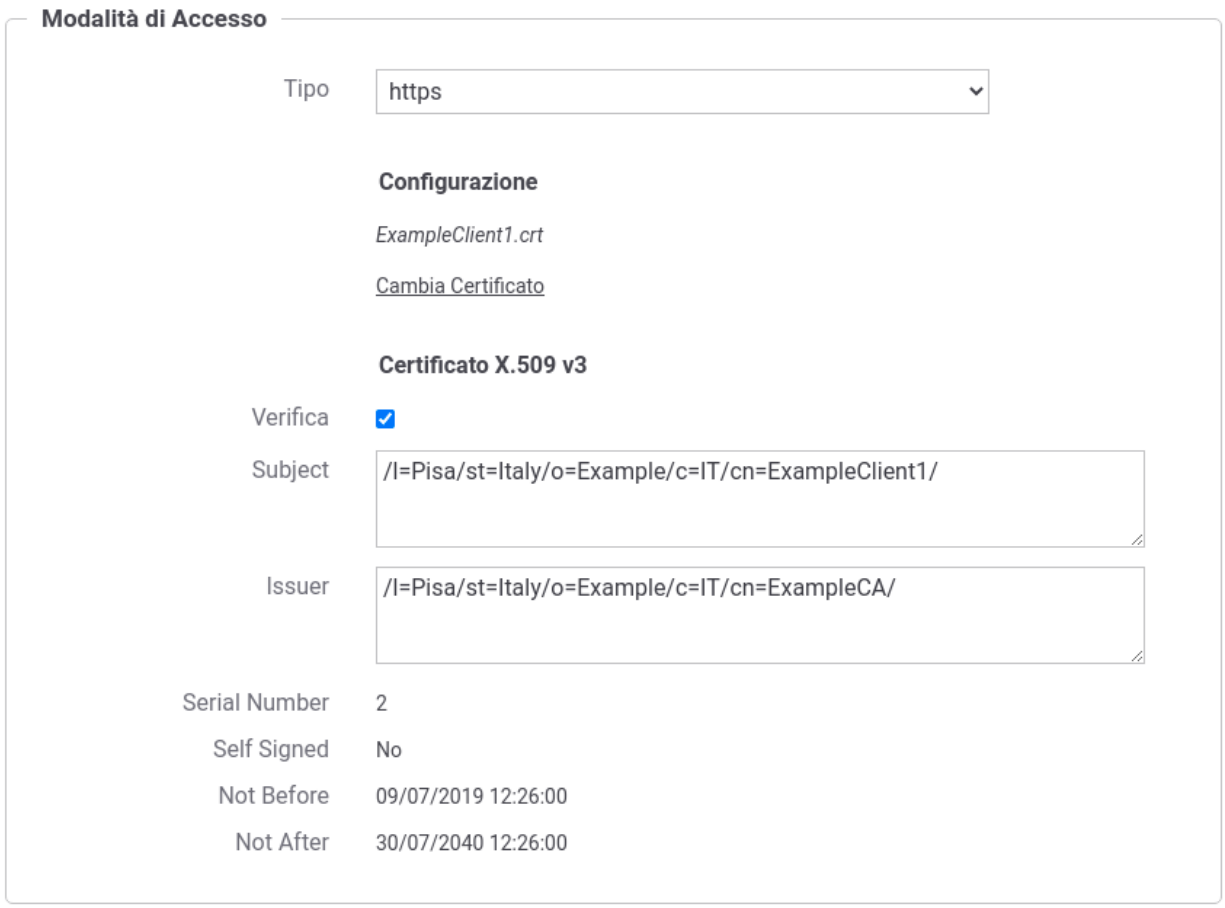


Fig. 2.78: Credenziali di tipo HTTPS (upload archivio 2/2)

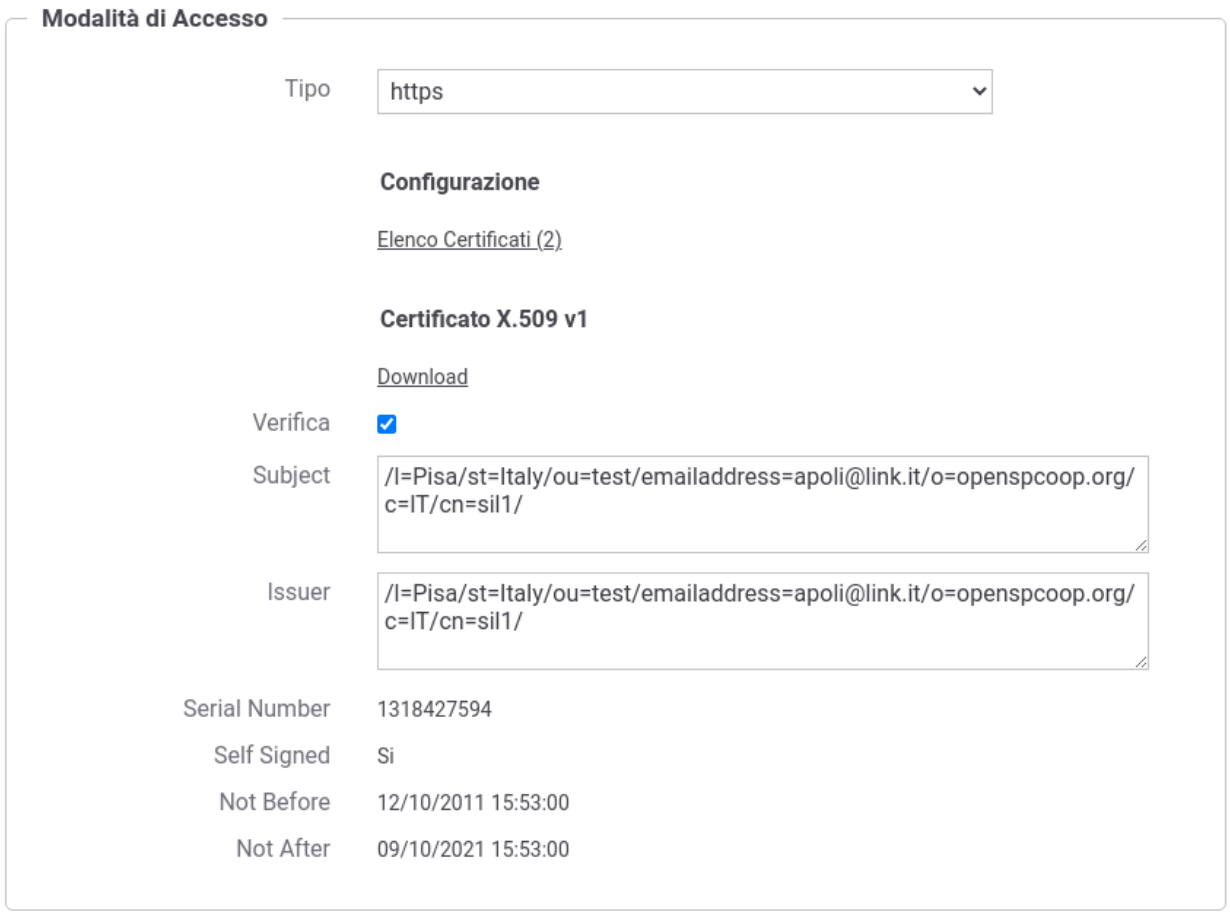


Fig. 2.79: Credenziali di tipo HTTPS (consultazione)

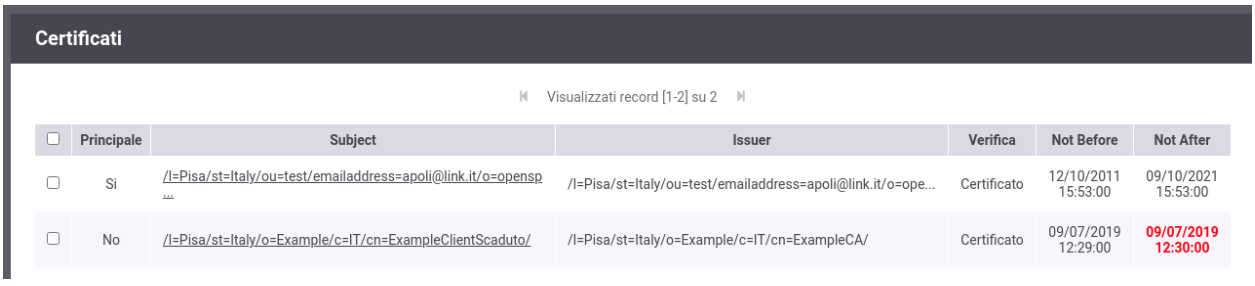


Fig. 2.80: Credenziali di tipo HTTPS (certificati aggiuntivi)

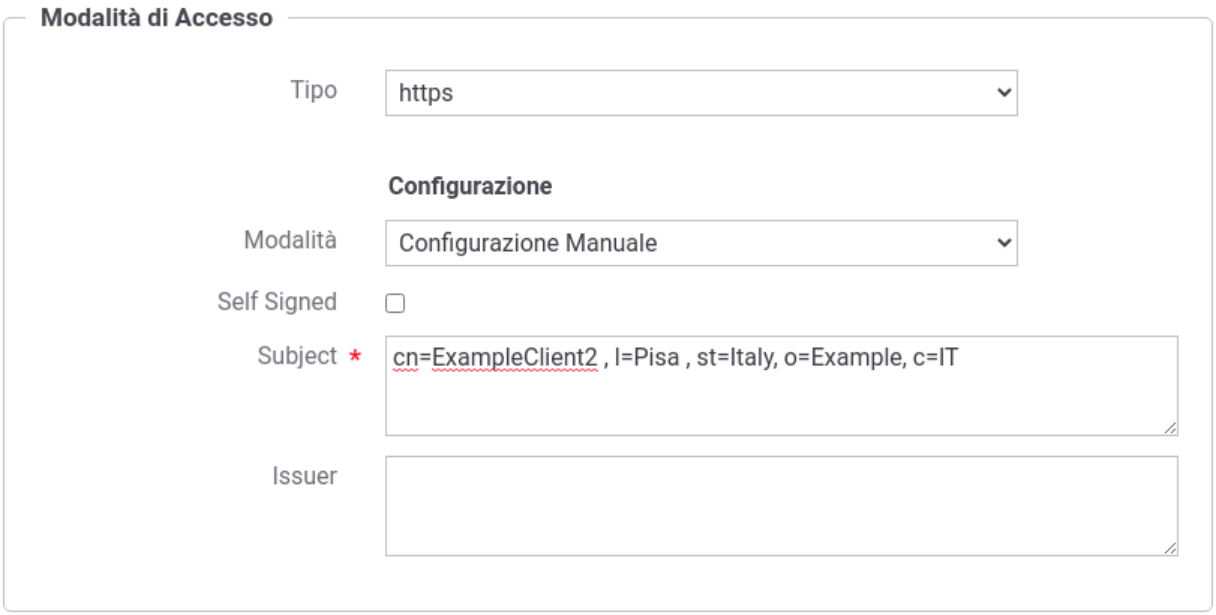


Fig. 2.81: Credenziali di tipo HTTPS (configurazione manuale)

Credenziali "http-basic"

Agli applicativi ed ai soggetti registrati nel gateway, identificabili con credenziali "http-basic", deve essere associato un identificativo utente univoco e una password (Fig. 2.82). La password può essere generata tramite l'apposito pulsante.

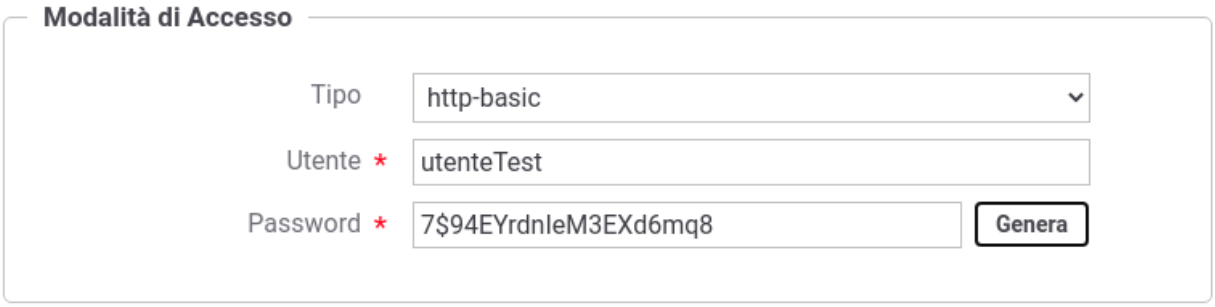


Fig. 2.82: Credenziali di tipo HTTP-Basic

Nota: La password generata e assegnata all'applicativo o al soggetto viene visualizzata solamente nell'avviso visualizzato in seguito alla creazione (Fig. 2.83) e successivamente non è più consultabile.

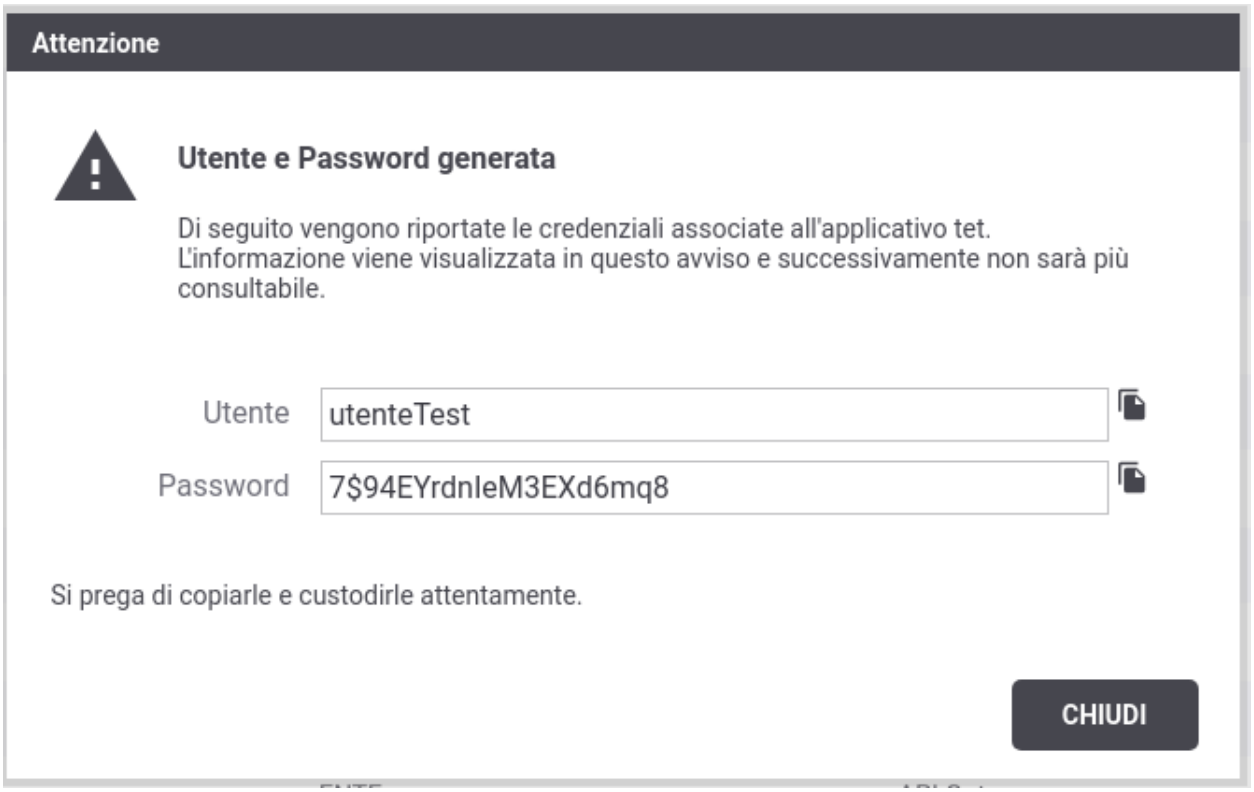


Fig. 2.83: Avviso di copia delle credenziali HTTP-Basic

Nel caso di smarrimento della password è necessario procedere con la generazione di una nuova password (Fig. 2.84).

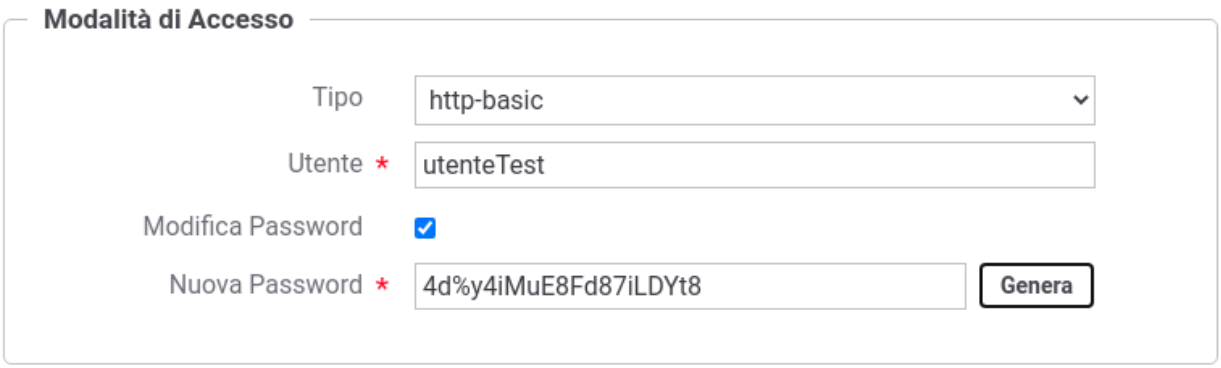


Fig. 2.84: Aggiornamento delle credenziali HTTP-Basic

Credenziali “api-key”

Agli applicativi ed ai soggetti registrati nel gateway, identificabili con credenziali “api-key” deve essere associato una chiave di identificazione univoca “Api Key” come descritto nella specifica “OAS3 API Keys” (https://swagger.io/docs/specification/authentication/api-keys/).

La credenziale può inoltre essere composta da un’ulteriore informazione riguardante l’identificatore dell’applicativo “App ID”; modalità denominata “Multiple API Keys” nella specifica “OAS3 API Keys”.

L’associazione di credenziali “api-key” ad un applicativo o soggetto comporta solamente l’indicazione se deve essere generato anche un “App ID” o meno (Fig. 2.85).

La generazione della “Api Key” e dell’eventuale “App ID” è automatica e viene visualizzata non appena si completa la registrazione dell’applicativo o del soggetto. Nella figura Fig. 2.86 viene riportato un avviso di generazione di una credenziale senza “App ID”, mentre nella figura Fig. 2.87 è stato generato anche l’identificatore dell’applicativo.

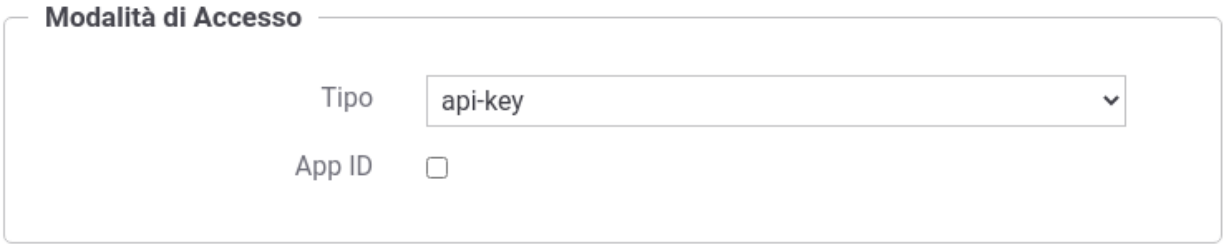


Fig. 2.85: Credenziali “api-key”

Nota: La chiave di identificazione generata e assegnata all’applicativo o al soggetto viene visualizzata solamente nell’avviso visualizzato in seguito alla creazione (Fig. 2.86) e successivamente non è più consultabile.

Nel caso di smarrimento della chiave è necessario procedere con la generazione di una nuova chiave (Fig. 2.88).

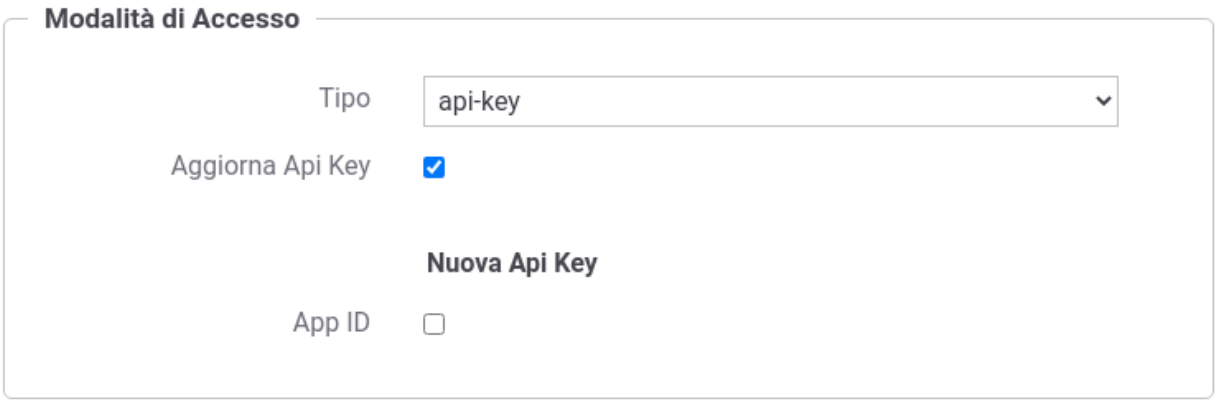


Fig. 2.88: Aggiornamento delle credenziali “api-key”

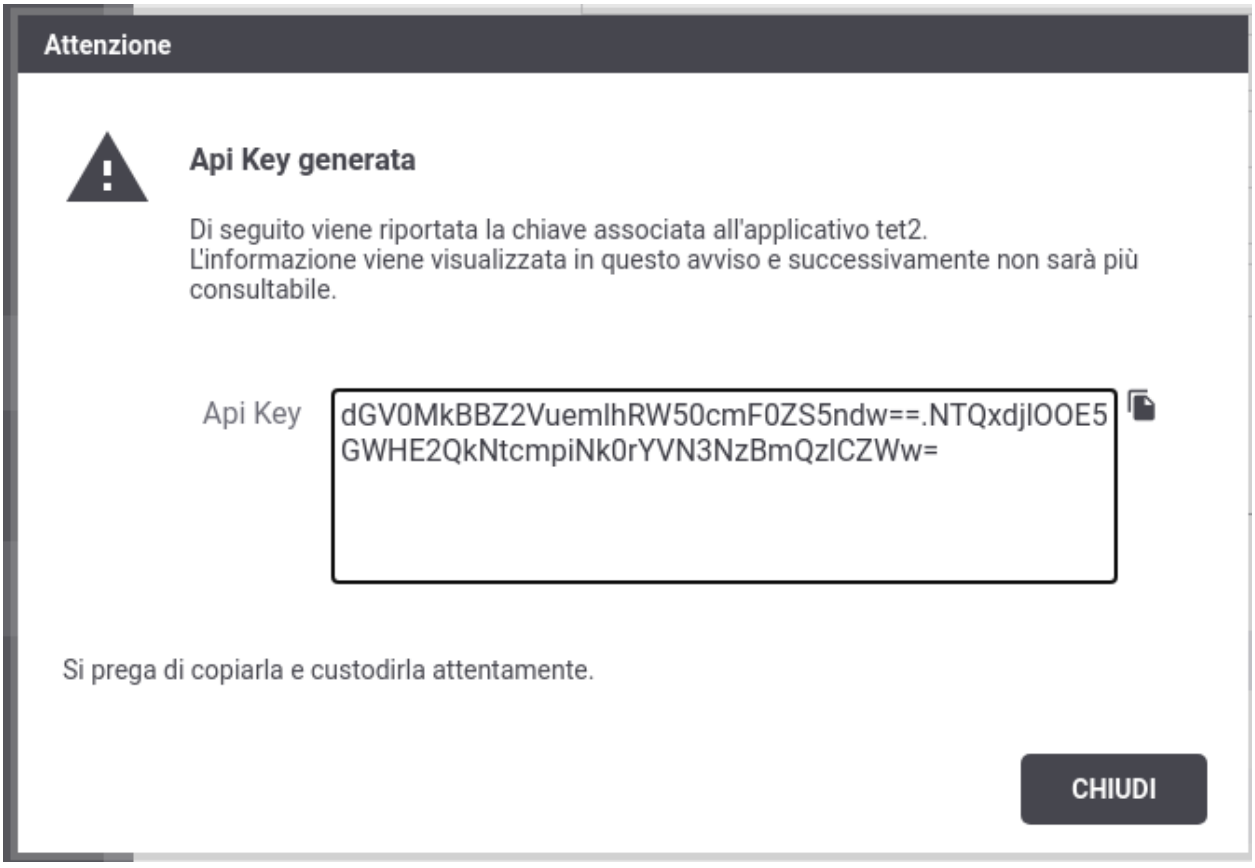


Fig. 2.86: Avviso di copia delle credenziali “api-key”

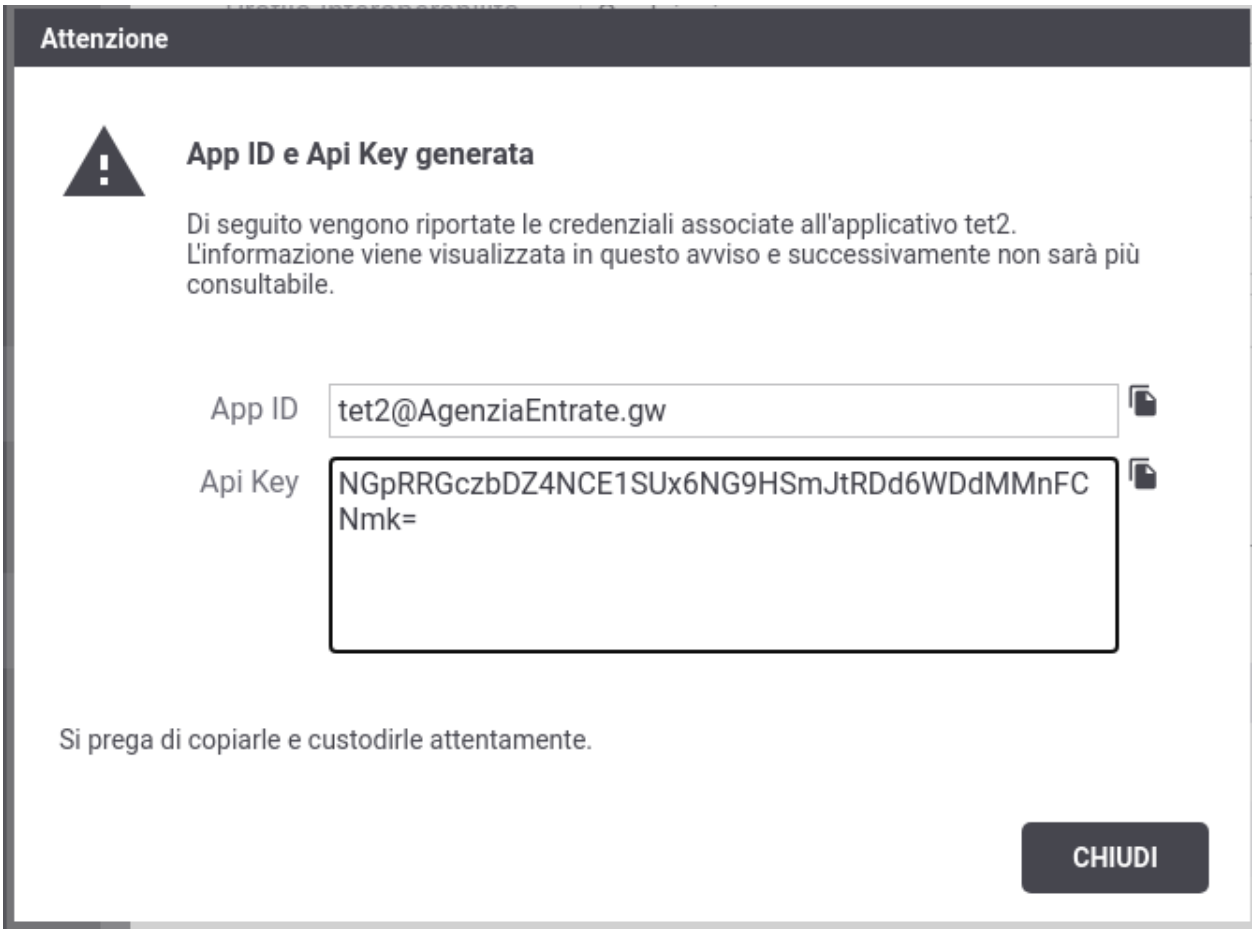


Fig. 2.87: Avviso di copia delle credenziali “api-key” (con App ID)

Credenziali "principal"

Agli applicativi ed ai soggetti registrati nel gateway, identificabili con credenziali "principal", deve essere associato un identificativo univoco (Fig. 2.89).

Fig. 2.89: Credenziali Principal

2.10.9 Creazione di un ruolo

È possibile censire i ruoli che potranno essere utilizzati come criterio di autorizzazione. Quelli contrassegnati come *fonte registro* potranno essere associandoli ai soggetti. Quelli invece contrassegnati come *fonte esterna* verranno assegnati dinamicamente ai soggetti che si autenticano, sulla base di quanto comunicato dal container dopo che l'utente ha effettuato l'autenticazione esternamente.

Per creare un nuovo ruolo ci si posiziona nella sezione *Registro* > *Ruoli* e si preme il pulsante *Aggiungi*.

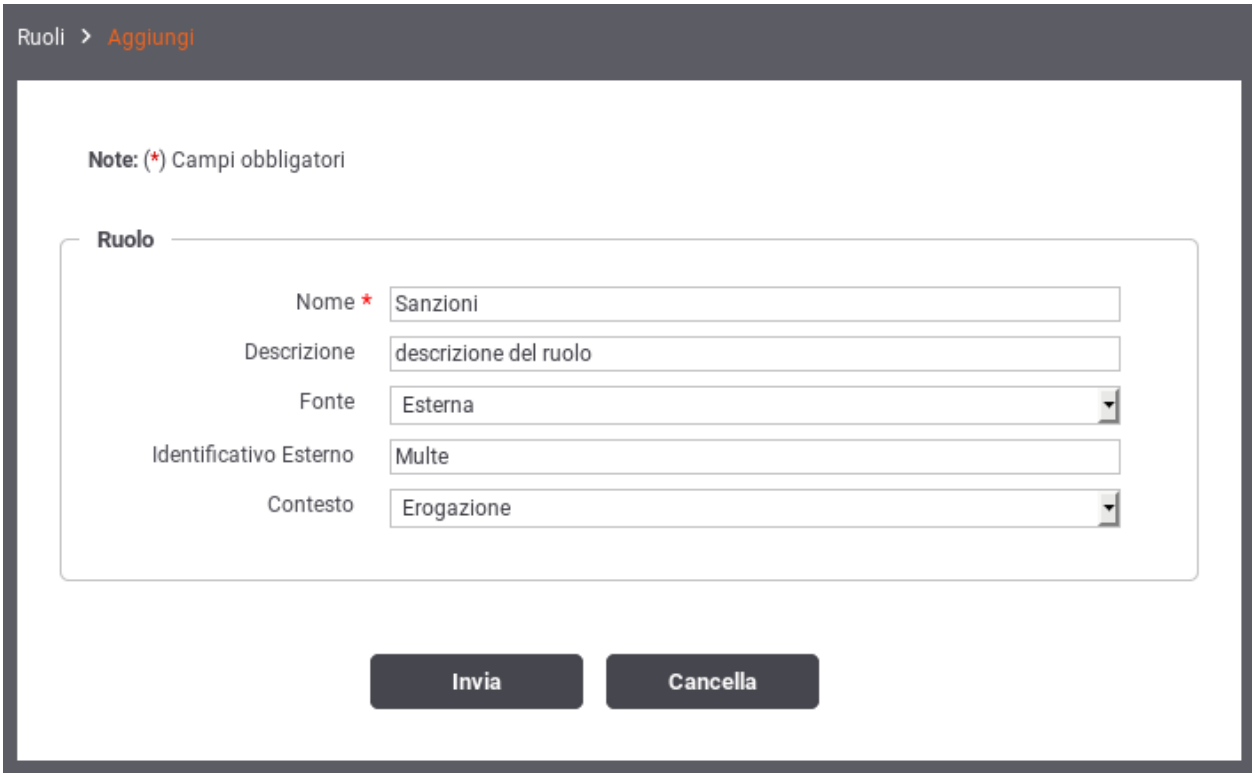


Fig. 2.90: Registrazione di un ruolo

Compilare il form (Fig. 2.90) nel seguente modo:

- *Nome*: identifica univocamente il ruolo.
- *Descrizione*: rappresenta una descrizione generica del ruolo.
- *Fonte*: la gestione del ruolo può essere effettuata direttamente su GovWay (fonte: registro) dove può essere assegnato ad un soggetto o applicativo. In alternativa (fonte: esterna) la gestione può essere delegata all'Application Server o a qualunque altra modalità che permetta al gateway di accedere ai ruoli tramite la api *HttpServletRequest.isUserInRole()*. In questo caso il nome del ruolo deve corrispondere allo stesso identificativo utilizzato nella configurazione esterna.

  Se non viene specificata alcuna fonte il ruolo potrà essere utilizzato per entrambe le modalità.
- *Contesto*: l'utilizzo del ruolo può essere limitato ad un contesto di erogazione o fruizione di servizio attraverso questa opzione.
- *Identificativo Esterno*: Nei casi in cui il ruolo provenga da un sistema esterno, è possibile che il suo identificativo sia differente rispetto a quello indicato nel contesto del Registro. In tal caso inserire in questo campo tale identificativo esterno.

2.10.10 Attribuzione dei Ruoli a Soggetti ed Applicativi

È possibile attribuire un ruolo ad un soggetto cliccando sulla voce "Ruoli" presente sia nell'elenco dei soggetti che nel dettaglio di un singolo soggetto. L'attribuzione consiste nello scegliere uno dei ruoli selezionabili per il soggetto tra quelli compatibili con il contesto di erogazione di servizio e che prevedono una fonte di registrazione interna al registro.

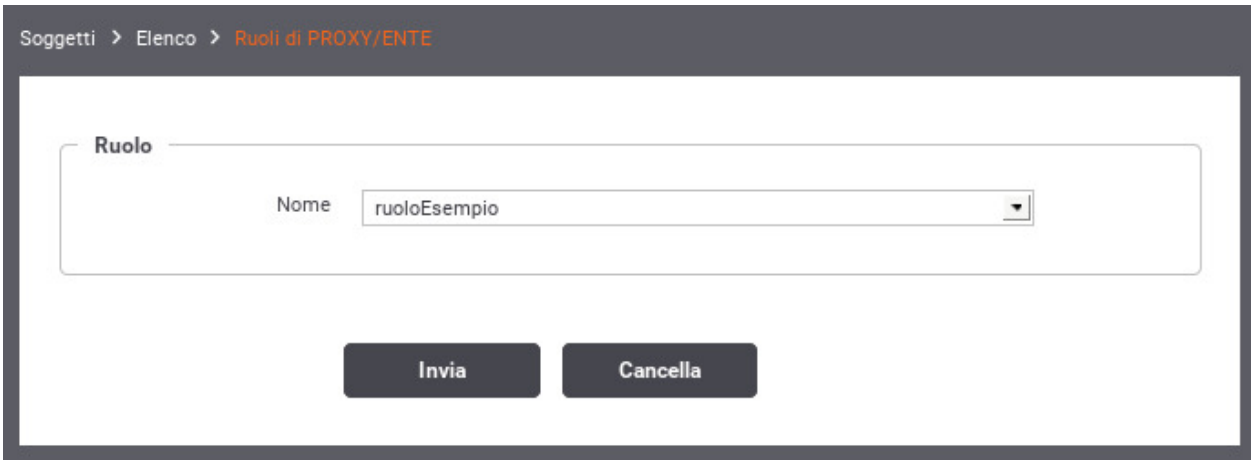


Fig. 2.91: Attribuzione di un ruolo ad un soggetto

In uguale maniera è possibile attribuire un ruolo ad un applicativo di tipologia *Frutore* cliccando sulla voce "Ruoli" presente nel dettaglio dell'applicativo. L'attribuzione consiste nello scegliere uno dei ruoli selezionabili per il servizio applicativo tra quelli compatibili con il contesto di fruizione di servizio e che prevedono una fonte di registrazione interna al registro.

Fig. 2.92: Attribuzione di un ruolo ad un applicativo

2.10.11 Autorizzazione per Token Claims

Se è stata abilitata la gestione del token si ha la possibilità di autorizzare le richieste inserendo i valori ammessi per i claims contenuti nel token.

L'autorizzazione per token claims permette di effettuare dei semplici controlli sui valori dei claim presenti nel token, una volta verificato che il token sia valido. La funzionalità è utilizzabile nei contesti in cui il controllo di autorizzazione possiede una logica semplice che si basa sulla verifica del valore di uno o più claim. Dove serve una logica più complessa (ad es. con rami "if-else") il controllo deve essere effettuato utilizzando una XACMLPolicy (*XACML-Policy*).

La configurazione viene effettuata inserendo nel campo di testo un claim da verificare per ogni riga, facendo seguire dopo l'uguale un valore fornito in una delle seguenti modalità:

- ${anyValue} : indica qualsiasi valore non nullo
- ${undefined} : la risorsa indicata non deve esistere o non deve essere valorizzata
- ${regExpMatch:EXPR} : la regola è soddisfatta se l'intero valore del claim ha un match rispetto all'espressione regolare EXPR indicata
- ${regExpFind:EXPR} : simile alla precedente regola, il match dell'espressione regolare può avvenire anche su una sottostringa del valore del claim
- valore : indica esattamente il valore (case sensitive) che deve possedere il claim; il valore può essere definito come costante o contenere parti dinamiche risolte a runtime dal Gateway descritte di seguito
- valore1,..,valoreN : è possibile elencare differenti valori ammissibili; come per la precedente opzione il valore può contenere parti dinamiche

Le espressioni utilizzabili come parti dinamiche, risolte a runtime dal gateway, sono:

- ${header:NAME}: valore presente nell'header http che possiede il nome "NAME"
- ${query:NAME}: valore associato al parametro della url con nome "NAME"
- ${urlRegExp:EXPR}: espressione regolare applicata sulla url di invocazione (l'espressione deve avere un match con l'intera url)
- ${xPath:EXPR}: espressione XPath applicata su un messaggio XML
- ${jsonPath:EXPR}: espressione JSONPath applicata su un messaggio JSON

- ${transportContext:FIELD}: permette di accedere ai dati della richiesta http; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.utils.transport.http.HttpServletTransportRequestContext" (es. per il principal usare ${transportContext:credential.principal})
- ${config:NAME}: valore della proprietà configurata sull'API che possiede il nome "NAME"
- ${clientApplicationConfig:NAME}: valore della proprietà configurata nell'applicativo fruitore che possiede il nome "NAME"
- ${clientOrganizationConfig:NAME}: valore della proprietà configurata nel soggetto fruitore che possiede il nome "NAME"
- ${providerOrganizationConfig:NAME}: valore della proprietà configurata nel soggetto erogatore che possiede il nome "NAME"
- ${system:NAME}: valore associato alla proprietà di sistema, indicata nella configurazione generale, con nome "NAME"
- ${env:NAME}: valore associato alla variabile di sistema con nome "NAME"
- ${java:NAME}: valore associato alla variabile java con nome "NAME"

Di seguito alcuni esempi:

- client_id=3 : viene verificato che il claim "client_id" possieda il valore 3
- client_id=3,5,6 : viene verificato che il claim "client_id" possieda il valore 3 o 5 o 6
- client_id=${anyValue} : viene verificato che il claim "client_id" possieda un valore (not null e not empty)
- client_id=${regExpMatch:[0-9]} : viene verificato che il claim "client_id" possieda esattamente una cifra decimale attraverso la verifica di un match con l'espressione regolare "[0-9]"
- client_id=${regExpFind:[0-9]} : viene verificato che il claim "client_id" contenga una cifra decimale attraverso l'espressione regolare "[0-9]"
- client_id=${header:X-Prova} : viene verificato che il claim "client_id" possieda lo stesso valore presente nell'header http "X-Prova" presente nella richiesta
- client_id=cl-${header:X-Prova} : viene verificato che il claim "client_id" possieda il valore presente nell'header http "X-Prova" arricchito del prefisso "cl-"
- client_id=${query:prova} : viene verificato che il claim "client_id" possieda lo stesso valore presente nel parametro "prova" della url di invocazione
- client_id=${urlRegExp:EXPR} : viene verificato che il claim "client_id" possieda lo stesso valore estratto dalla url di invocazione attraverso l'applicazione dell'espressione regolare EXPR
- client_id=${xPath:EXPR} : viene verificato che il claim "client_id" possieda lo stesso valore estratto dalla richiesta xml tramite l'espressione XPath EXPR.
- client_id=${jsonPath:EXPR} : viene verificato che il claim "client_id" possieda lo stesso valore estratto dalla richiesta json tramite l'espressione jsonPath EXPR.

Per verificare un attributo indicarlo con il prefisso "attribute." nella forma "attribute.nome=valore". Di seguito alcuni esempi

- attribute.sesso=m : viene verificato che l'attributo "sesso" possieda il valore m
- attribute.stato=3,5,6 : viene verificato che l'attributo "stato" possieda il valore 3 o 5 o 6

Nel caso la configurazione relativa all'*Identificazione Attributi* prevede più AA, la verifica di un attributo prelevato da un authority va indicato con i prefissi "aa." e "attribute." nella forma "aa.nomeAuthority.attribute.nomeAttributo=valore".

- aa.AA2.attribute.sesso=m : viene verificato che l'attributo "sesso", prelevato tramite l'Attribute Authority "AA2", possia il valore m
- aa.AA2.attribute.stato=3,5,6 : viene verificato che l'attributo "stato", prelevato tramite l'Attribute Authority "AA2", possia il valore 3 o 5 o 6

2.10.12 XACML-Policy

Questa tipologia di autorizzazione prevede di limitare l'accesso ai soli applicativi o soggetti fruitori che soddisfino una determinata policy XACML. La policy deve essere caricata nel contesto dell'autorizzazione sul controllo degli accessi, come mostrato in Fig. 2.93.

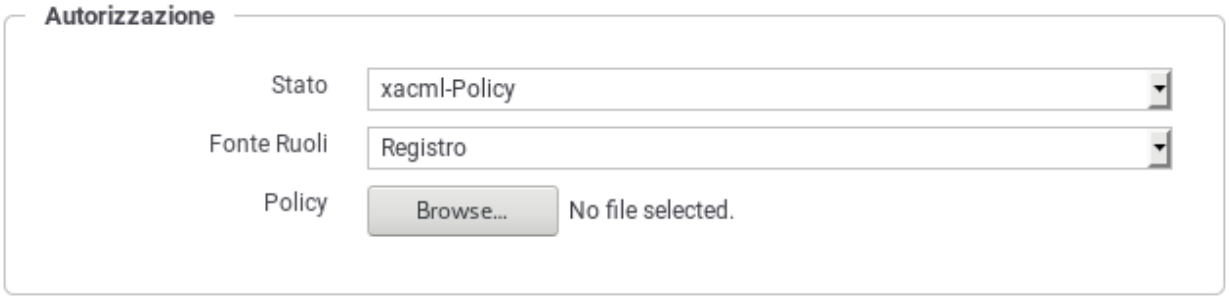


Fig. 2.93: Registrazione di una XACML-Policy per l'erogazione

In fase di autorizzazione, il gateway costruisce una XACMLRequest contenente tutti i parametri della richiesta, comprese le informazioni relative al chiamante (credenziali ed eventuali ruoli), e la valida rispetto alla XACML-Policy associata all'erogazione. I parametri inseriti nella XACMLRequest, che possono essere utilizzati per effettuare la verifica all'interno di una XACML-Policy, sono i seguenti:

Tabella 2.1: Parametri inseriti in una XACMLRequest

Nome	Descrizione
Sezione "Action"	
org:govway:action:provider	Indica il soggetto erogatore del servizio
org:govway:action:provider:config:<nome>	Proprietà configurate nel soggetto erogatore del servizio
org:govway:action:service	Indica il servizio nel formato tipo/nome
org:govway:action:service:config:<nome>	Proprietà configurate nell'erogazione o nella fruizione
org:govway:action:action	Nome dell'operazione del servizio invocata
org:govway:action:url	Url di invocazione utilizzata dal mittente
org:govway:action:url:parameter:NOME_PARAM	Tutti i parametri presenti nell'url di invocazione saranno inseriti nella XACMLRequest con questo formato
org:govway:action:transport:header:NOME_HDR	Tutti gli header http presenti nell'url di invocazione saranno inseriti nella XACMLRequest con questo formato
org:govway:action:soapAction	Valore della SOAPAction
org:govway:action:gwService	Ruolo della transazione (inbound/outbound)
org:govway:action:protocol	Modalità associata al servizio richiesto (es. spcoop)

continues on next page

Tabella 2.1 – continua dalla pagina precedente

Nome	Descrizione
org:govway:action:token:audience	Destinatario del token
org:govway:action:token:scope	Lista di scopes
org:govway:action:token:jwt:claim:<nome>	Tutti i claims presenti nel jwt validato
org:govway:action:token:introspection:claim:<nome>	Tutti i claims presenti nella risposta del servizio di introspection
Sezione "Subject"	
org:govway:subject:organization	Indica il soggetto fruitore
org:govway:subject:organization:config:<nome>	Proprietà configurate nel soggetto fruitore
org:govway:subject:client	Identificativo dell'applicativo client
org:govway:subject:client:config:<nome>	Proprietà configurate nell'applicativo client
org:govway:subject:credential	Rappresenta la credenziale di accesso (username, subject o il principal) utilizzata dal client per richiedere il servizio
org:govway:subject:role	Elenco dei ruoli che possiede il client che ha richiesto il servizio
org:govway:subject:token:issuer	Issuer del token
org:govway:subject:token:subject	Subject del token
org:govway:subject:token:username	Username dell'utente cui è associato il token
org:govway:subject:token:clientId	Identificativo del client che ha negoziato il token
org:govway:subject:token:userInfo:fullName	Nome completo dell'utente cui è associato il token
org:govway:subject:token:userInfo:firstName	Nome dell'utente cui è associato il token
org:govway:subject:token:userInfo:middleName	Secondo nome (o nomi aggiuntivi) dell'utente cui è associato il token
org:govway:subject:token:userInfo:familyName	Cognome dell'utente cui è associato il token
org:govway:subject:token:userInfo:eMail	Email dell'utente cui è associato il token
org:govway:subject:token:userInfo:claim:<nome>	Tutti i claims presenti nella risposta del servizio di UserInfo
org:govway:subject:attributes	Elenco dei nomi degli attributi recuperati interagendo con gli Attribute Authority configurati
org:govway:subject:attribute:<nome>	In caso sia configurato un unico Attribute Authority, nella configurazione relativa all'*Identificazione Attributi*, tutti gli attributi recuperati saranno inseriti nella XACMLRequest con questo formato
org:govway:subject:aa:<attributeAuthority>:attribute:<nome>	In caso siano configurate più Attribute Authority, nella configurazione relativa all'*Identificazione Attributi*, tutti gli attributi recuperati saranno inseriti nella XACMLRequest con questo formato

Di seguito un esempio di XACMLPolicy che autorizza le richieste dei chiamanti che possiedono il ruolo "Amministratore" ed uno tra i due ruoli "Operatore1" e "Operatore2":

```
<Policy xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os" xmlns:xsi="http://www.
↪w3.org/2001/XMLSchema-instance" PolicyId="Policy" RuleCombiningAlgId=
↪"urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:permit-overrides"␣
↪xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os http://docs.oasis-
↪open.org/xacml/2.0/access_control-xacml-2.0-policy-schema-os.xsd">
```

(continues on next page)

(continua dalla pagina precedente)

```
   <Target />
   <Rule Effect="Permit" RuleId="ok">
      <Condition>
         <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
            <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-
↪one-member-of">
               <SubjectAttributeDesignator AttributeId="org:govway:subject:role"␣
↪DataType="http://www.w3.org/2001/XMLSchema#string" />
               <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
                 <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
↪Amministratore</AttributeValue>
               </Apply>
            </Apply>
            <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-
↪one-member-of">
               <SubjectAttributeDesignator AttributeId="org:govway:subject:role"␣
↪DataType="http://www.w3.org/2001/XMLSchema#string" />
               <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
                  <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
↪Operatore1</AttributeValue>
                  <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
↪Operatore2</AttributeValue>
               </Apply>
            </Apply>
         </Apply>
      </Condition>
    </Rule>
    <Rule Effect="Deny" RuleId="ko" />
</Policy>
```

Un altro esempio di policy che verifica l'uguaglianza tra il valore del claim "sub" presente nel token e quello fornito nel query parameter "sub" è la seguente:

```
<Policy PolicyId="Policy"
     RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:permit-
↪overrides"
     xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os" xmlns:xsi="http://www.w3.
↪org/2001/XMLSchema-instance"
     xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os http://docs.
↪oasis-open.org/xacml/2.0/access_control-xacml-2.0-policy-schema-os.xsd">
     <Target />
     <Rule Effect="Permit" RuleId="ok">
             <Condition>
                     <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:or">

                                <Apply FunctionId="urn:oasis:names:tc:xacml:1.
↪0:function:any-of-any">
                                        <Function FunctionId="urn:oasis:names:tc:xacml:1.
↪0:function:string-equal"/>
                                        <ActionAttributeDesignator
                                            AttributeId=
↪"org:govway:action:url:parameter:sub"
                                            DataType="http://www.w3.org/2001/XMLSchema
↪#string"
                                            MustBePresent="false"
                                        />
```

(continues on next page)

(continua dalla pagina precedente)

```
                                        <ActionAttributeDesignator
                                            AttributeId=
↪"org:govway:action:token:introspection:claim:sub"
                                            DataType="http://www.w3.org/2001/XMLSchema
↪#string"
                                            MustBePresent="false"
                                        />
                                    </Apply>

                            </Apply>
                    </Condition>
        </Rule>
        <Rule Effect="Deny" RuleId="ko" />
</Policy>
```

2.10.13 Scope

Nella sezione *Registro* > *Scope* è possibile gestire il censimento degli scope da utilizzare successivamente per le politiche di autorizzazione nell'ambito del controllo degli accessi.

La maschera di creazione di uno scope è quella mostrata in Fig. 2.94.

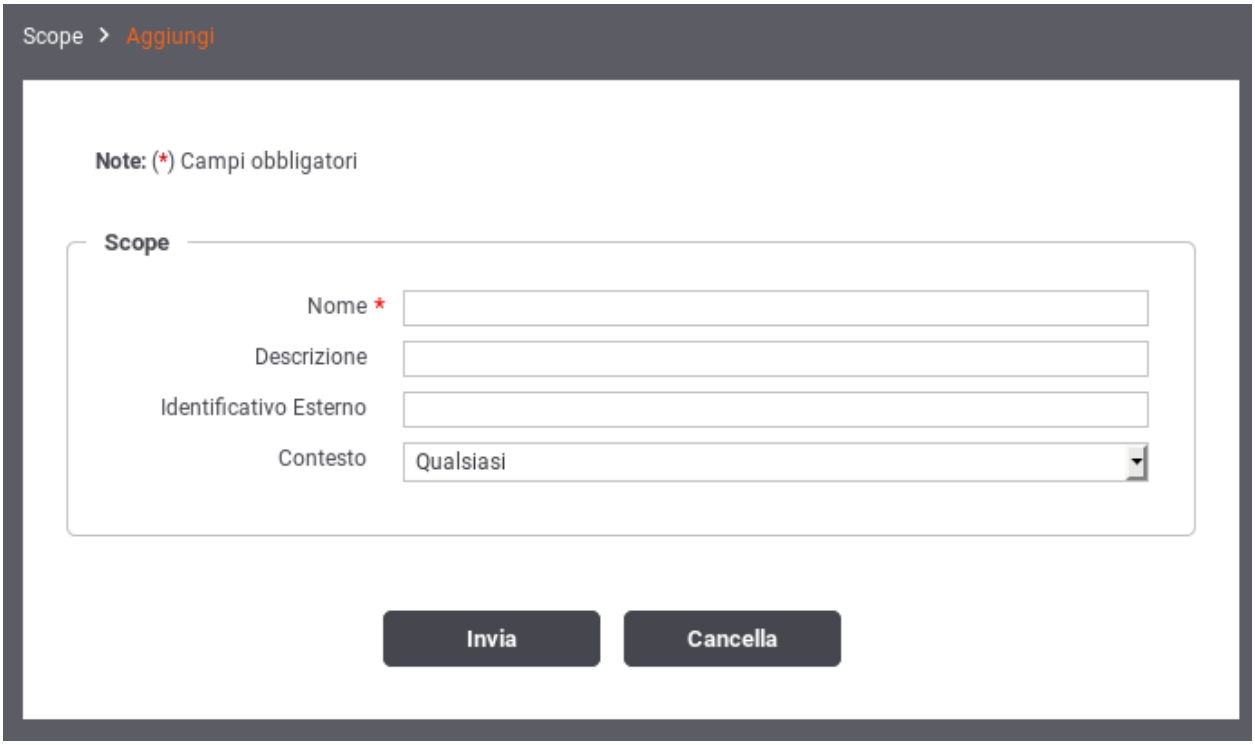


Fig. 2.94: Creazione di uno Scope

I dati da fornire sono:

- *Nome*: nome assegnato internamente allo scope
- *Descrizione*: un testo di descrizione

- *Identificativo Esterno*: nome originale dello scope presente nel token
- *Contesto*: specifica se lo scope si utilizza solo nei contesti di erogazione, fruizione o entrambe le possibilità.

2.11 Rate Limiting

Questa sezione di configurazione, specifica per erogazioni e fruizioni (o specifico gruppo di configurazione nell'ambito di un'erogazione/fruizione), consente di attivare delle policy di Rate Limiting specifiche per l'istanza configurata.

L'attivazione di policy di rate limiting rientra nell'ambito degli strumenti per il controllo del traffico. La descrizione di dettaglio di questi strumenti è presente nella sezione *Controllo del Traffico*, dove viene illustrato il meccanismo per configurare le policy e più in dettaglio nella sezione *Policy Globali* riguardo l'attivazione di policy a valenza globale.

Una policy di rate limiting si compone concettualmente dei seguenti elementi:

- *Criterio di Misurazione*: elemento che consente di calcolare un valore utile per la valutazione della policy. Il valore calcolato dipende dalla **metrica** scelta. La metrica viene scelta in fase di configurazione tra quelle disponibili, che sono:
  - *Numero Richieste*: consente di limitare il numero totale massimo di richieste consentite.
  - *Numero Richieste Simultanee*: limita il numero totale massimo di richieste simultanee consentite.
  - *Dimensione Massima Messaggi*: limita la dimensione massima accettata di una richiesta e di una risposta.
  - *Occupazione Banda*: limita il numero totale massimo di KB consentiti.
  - *Tempo Medio Risposta*: la policy blocca ogni successiva richiesta se viene rilevato un tempo medio di risposta elevato.
  - *Tempo Complessivo Risposta*: la policy limita il numero totale massimo di secondi consentiti.
  - *Numero Richieste Completate con Successo*: vengono conteggiate solamente il numero di richieste completate con successo; raggiunto il limite, ogni successiva richiesta viene bloccata.
  - *Numero Richieste Fallite*: vengono conteggiate il numero di richieste fallite; raggiunto il limite, ogni successiva richiesta viene bloccata.
  - *Numero Fault Applicativi*: vengono conteggiate il numero di richieste che veicolano un fault applicativo; raggiunto il limite, ogni successiva richiesta viene bloccata.
  - *Numero Richieste Fallite o Fault Applicativi*: vengono conteggiate il numero di richieste fallite o che veicolano un fault applicativo; raggiunto il limite, ogni successiva richiesta viene bloccata.

  Per ottenere un valore di confronto, alla metrica è necessario associare un intervallo di osservazione che consente di stabilire univocamente il conteggio risultante (fa eccezione "Numero Richieste Simultanee"). L'intervallo di osservazione può essere espresso scegliendone uno tra i seguenti:
  - *Minuti*
  - *Orario*
  - *Giornaliero*
- *Soglia di Confronto*: elemento della policy che fornisce il valore di soglia da confrontare con il valore ottenuto dalla metrica impostata.
- *Filtro di Applicabilità*: elemento della policy che stabilisce i criteri per i quali è applicabile la policy sui flussi in elaborazione sul Gateway (filtro su mittente, api, applicativo, ecc.).

Per ogni singola erogazione o fruizione di API è possibile definire più politiche di Rate Limiting, anche con medesima metrica. Per ogni richiesta viene applicato un algoritmo di valutazione delle policy che è il seguente (una descrizione di dettaglio viene fornita nella sezioni successive):

- le policy vengono raggruppate «per metrica» e per ogni metrica vengono valutate nell'ordine di elenco.
- per ogni metrica vengono valutate le policy applicabili, cioè per le quali risultano soddisfatti il filtro e le condizioni di applicabilità.
- se la policy viola i livelli di soglia previsti, la transazione viene bloccata (o segnalata se configurata come «warning only») e la valutazione delle policy viene terminata.
- se la policy non viola invece i livelli di soglia previsti, si prosegue nella valutazione di ulteriori policy per quella metrica, solo se la policy è marcata come «prosegui».

2.11.1 Registrazione di una policy

Per attivare una nuova policy dalla sezione di rate limiting si procede utilizzando il pulsante *Aggiungi* che apre il form di Fig. 2.95.

Si compilano i campi seguenti:

- *Policy*: la policy da attivare. Si compone di:
  - *Nome*: Identificativo univoco della policy.
  - *Stato*: Lo stato della policy. Sono disponibili le seguenti opzioni:
    - *Abilitato*: le violazioni rilevate saranno gestite in maniera restrittiva (negazione del servizio).
    - *WarningOnly*: la policy è abilitata in modalità WarningOnly. Questo significa che le violazioni rilevate saranno solo segnalate tramite messaggi diagnostici ma non ci saranno ripercussioni sull'elaborazione della richiesta.
    - *Disabilitato*: La policy è disabilitata.
  - *Elaborazione*: Indica quale azione attuare per la policy, nell'ambito del flusso di elaborazione delle policy di eguale metrica, nel caso in cui venga superato il controllo (maggiori dettagli sull'algoritmo di valutazione delle policy sono disponibili nella sezione *Criteri di valutazione delle policy*):
    - *Interrompi*: non verranno valutate ulteriori policy che seguono nell'ordine tra quelle di eguale metrica.
    - *Prosegui*: si procede con la valutazione della successiva policy nell'ordine tra quelle di eguale metrica.
  - *Identificazione Policy*: Scelta tra due opzioni:
    - *Scegli Criteri*: permette di indicare direttamente i criteri che la politica deve garantire; tra i criteri utilizzabili: la metrica (numero richieste, occupazione banda, tempi medi, ...), l'intervallo temporale (minuto, ora, giorno) e le condizioni di applicabilità (congestione, degrado prestazionale).
    - *Selezione dal Registro*: permette di utilizzare una politica arbitraria, precedentemente definita dall'utente.

    Nota: La descrizione che segue assume che venga attuata una identificazione della policy per criteri. Per i dettagli sulla configurazione di policy personalizzate dall'utente si faccia riferimento alla sezione *Rate Limiting*.
  - *Criteri*: devono essere forniti la metrica e l'intervallo di osservazione scelti tra i valori descritti in precedenza (*Rate Limiting*). Possono inoltre essere indicate le seguenti opzioni:

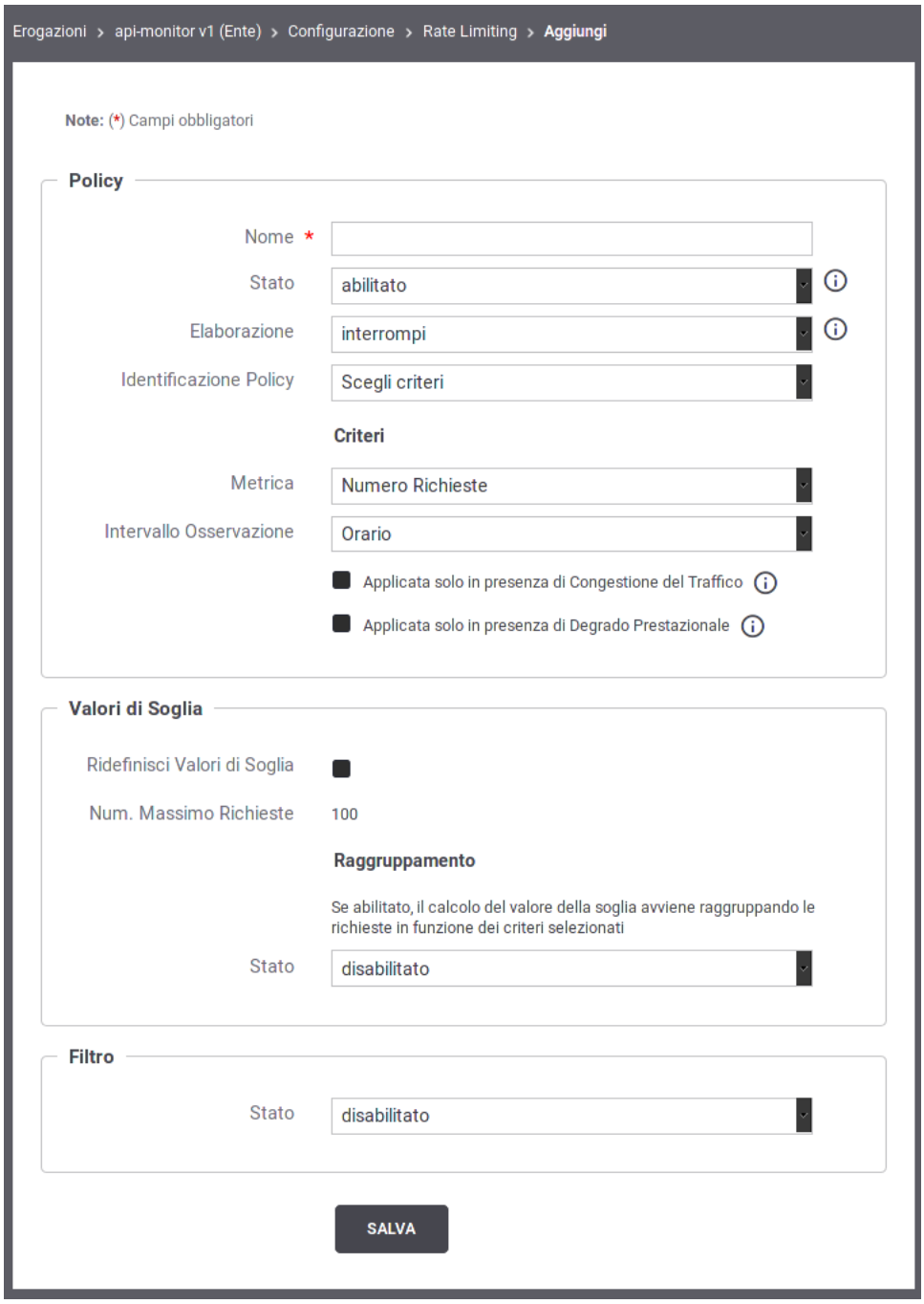


Fig. 2.95: Attivazione di una policy di Rate Limiting

        * *Applicata solo in presenza di Congestione del Traffico*: attivando questa opzione la policy risulta applicabile solo nel caso in cui il gateway sia entrato in modalità «Congestione», sulla base di quanto descritto nella sezione *Controllo del Traffico*.
        * *Applicata solo in presenza di Degrado Prestazionale*: attivando questa opzione la policy risulta applicabile solo nel caso in cui il gateway abbia rilevato un degrado prestazionale e cioè un tempo medio di risposta del servizio superiore alla soglia configurata.

- *Valori di Soglia*: Le soglie per la valutazione della policy:
    - *Ridefinisci Valori di Soglia*: Opzione che consente di variare la soglia predefinita.
    - *Soglia*: Questo campo riporta, in base alla metrica selezionata sopra, il valore di riferimento. Tale valore risulta modificabile attivando l'opzione al punto precedente.
    - *Raggruppamento*: In questa sezione è possibile attivare opzionalmente alcuni criteri per il raggruppamento dei dati utilizzati come soglie di confronto. Ad esempio se la policy limita a 20 il numero di richieste su base per minuti, significa che al raggiungimento della ventunesima richiesta, nella stessa finestra temporale, si otterrà una violazione della policy. Aggiungendo un raggruppamento per risorsa, saranno conteggiate separatamente le richieste in base alla specifica risorsa invocata. In questo caso la policy risulterà violata solo al raggiungimento della ventunesima richiesta, nella stessa finestra temporale, relativa alla medesima risorsa. È ammesso anche il raggruppamento su criteri multipli. La logica è del tutto analoga a quella dell'operatore GROUP BY del linguaggio SQL. I criteri di raggruppamento selezionabili sono:
        * *Risorsa/Azione*: il valore di soglia rappresenta il totale per ciascuna azione/risorsa
        * *Richiedente*: il valore di soglia rappresenta il totale ripartito per ciascun mittente
        * *Token*: il valore di soglia rappresenta il totale ripartito tra le richieste in base al token di provenienza. Si possono specificare i «claims» da prendere in considerazione per distinguere i token.
        * *Chiave*: il valore di soglia rappresente il totale ripartito tra le richieste raggruppate in base ad una chiave personalizzata il cui valore viene fornito secondo uno dei metodi selezionati tra i seguenti:
            · *Header HTTP*: La chiave è presente nell'header di trasporto indicato nella proprietà «Nome».
            · *Url di Invocazione*: La chiave è presente nella URL ricavabile tramite l'espressione regolare fornita nell'elemento seguente (l'espressione deve avere un match con l'intera url).
            · *Parametro della Url*: La chiave viene fornita in modalità Form Encoded con il parametro indicato nell'elemento «Nome».
            · *SOAPAction*: La chiave corrisponde al valore della SoapAction.
            · *Contenuto*: La chiave è presente nel body del messaggio e viene ricavata tramite una espressione XPath o JsonPath fornito nell'elemento seguente.
            · *Client IP*: La chiave corrisponde all'indirizzo IP del client.
            · *X-Forwarded-For*: La chiave corrisponde all'indirizzo IP del client presente negli header http utilizzati per il mantenimento dell'IP di origine nel caso di nodi intermedi (es. X-Forwarded-For).
            · *Plugin Personalizzato*: La chiave viene restituita tramite l'esecuzione di una classe il cui nome viene fornito con il campo «Tipo». Per maggiori dettagli si rimanda alla sezione *Filtro o Raggruppamento Personalizzato*

- *Filtro*: Abilitando questa sezione è possibile indicare i criteri affinché la policy sia applicabile in base alle caratteristiche di ciascuna richiesta in ingresso. In assenza di filtro, la policy sarà valutata su tutte le richieste in ingresso che riguardano l'erogazione/fruizione che si sta configurando. Per la creazione del filtro sono disponibili i seguenti campi:
    - *Risorsa/Azione*: Opzione per filtrare le richieste in base all'azione/risorsa invocata.

- *Ruolo Richiedente*: Opzione per filtrare le richieste in base al ruolo posseduto dal richiedente (sia che si tratti di un soggetto che di un applicativo).
- *Soggetto o Applicativo Fruitore*: In aternativca al filtro per ruolo, è possibile specificare un soggetto fruitore ed eventualmente uno dei suoi applicativi.
- *Chiave*: Si tratta di un'opzione avanzata che consente di filtrare le richieste in ingresso sul gateway in base ad una chiave che può essere specificata in maniera personalizzata effettuando una delle seguenti scelte per il campo *Tipologia*:
  - *Header HTTP*: Occorre fornire i dati "Nome" e "Valore". La policy si applicherà soltanto alle richieste che hanno un header http che corrisponde.
  - *Url di Invocazione*: Occorre fornire i dati "Espressione Regolare" e "Valore". La policy si applicherà soltanto alle richieste ove, applicando l'espressione regolare alla URL di invocazione, si ottiene un valore identico a quello fornito.
  - *Parametro della Url*: Occorre fornire i dati "Nome" e "Valore". La policy si applicherà soltanto alle richieste che contengono nella url di invocazione un parametro corrispondente ai dati forniti.
  - *SOAPAction*: Occorre fornire il dato "Valore". La policy si applicherà soltanto alle richieste che si presentano con una SOAPAction avente il valore fornito.
  - *Contenuto*: Occorre fornire i dati "Pattern" e "Valore". La policy si applicherà soltanto alle richieste dove, applicando l'espressione XPath o JsonPath al messaggio di richiesta, si ottiene un valore identico a quello fornito.
  - *Client IP*: La policy si applicherà soltanto alle richieste che provengono dall'indirizzo IP indicato.
  - *X-Forwarded-For*: La policy si applicherà soltanto alle richieste che provengono dall'indirizzo IP indicato presente negli header http utilizzati per il mantenimento dell'IP di origine nel caso di nodi intermedi (es. X-Forwarded-For).
  - *Plugin Personalizzato*: Permette di definire un criterio di filtro personalizzato. Per maggiori dettagli si rimanda alla sezione *Filtro o Raggruppamento Personalizzato*

2.11.2 Criteri di valutazione delle policy

Le policy di rate limiting create, per la data erogazione/fruizione, sono visualizzate in un elenco che filtra automaticamente su una singola metrica (ad esempio «Numero Richieste» o «Occupazione Banda»). L'elenco delle policy visualizzato è analogo a quello riportato in Fig. 2.96.

Ciascun elemento in elenco riporta le seguenti informazioni:

- *Ordine*: pulsanti per variare la posizione della policy nell'elenco per la data metrica.
- *Stato*: lo stato di abilitazione della policy, sulla base di quanto descritto in precedenza (*Registrazione di una policy*).
- *Nome*: il nome della policy.
- *Soglia*: il valore di soglia impostato per la policy.
- *Runtime*: permette di effettuare una verifica in tempo reale della metrica interrogando il runtime del gateway. Maggiori dettagli sono presenti nella sezione *Visualizzazione Statistiche Policy*.
- *Elaborazione*: flusso di elaborazione (prosegui, interrompi) nel caso di superamento del controllo relativo alla policy.

L'elenco delle policy può essere aggiornato utilizzando il meccanismo di filtro presente nell'intestazione della tabella. Sono disponibili le seguenti opzioni:

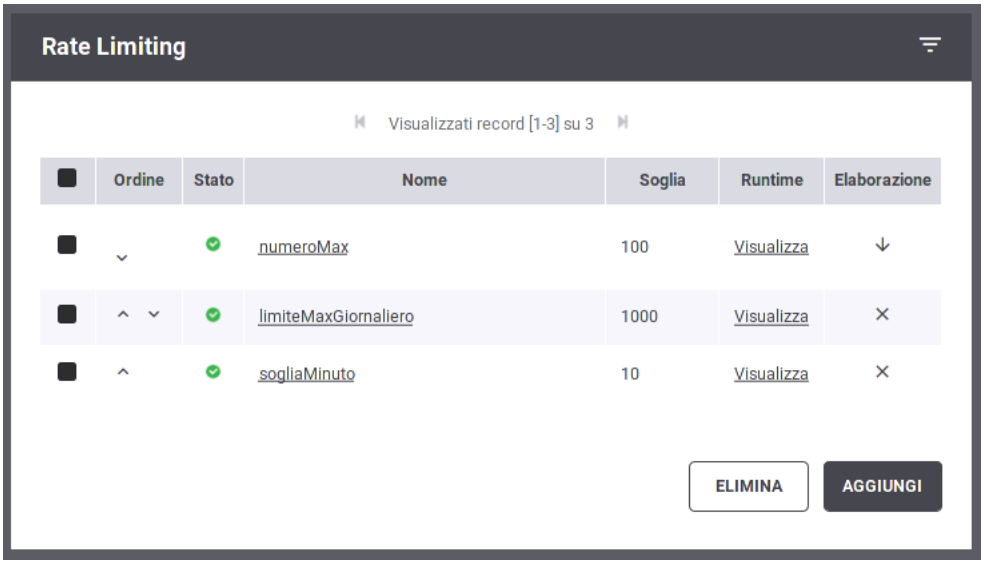


Fig. 2.96: Elenco delle policy di Rate Limiting

- *Metrica*: permette di stabilire le policy da visualizzare in base alla rispettiva metrica.
- *Ricerca*: permette di visualizzare le policy in base alla presenza di un pattern nel nome.

Per ogni richiesta relativa alla specifica erogazione/fruizione viene applicato l'algoritmo di valutazione delle policy che è il seguente:

- le policy vengono raggruppate «per metrica» e per ogni metrica vengono valutate nell'ordine di elenco prima utilizzando le politiche di Rate Limiting definite sull'API e poi, se esistenti, le politiche a valenza globale (*Policy Globali*).
- per ogni metrica vengono valutate le policy applicabili, cioè per le quali risultano soddisfatti il filtro e le condizioni di applicabilità.
- se la policy viola i livelli di soglia previsti, la transazione viene bloccata (o segnalata se configurata come «warning only») e la valutazione delle policy viene terminata.
- se la policy non viola invece i livelli di soglia previsti, si prosegue nella valutazione di ulteriori policy per quella metrica, solo se la policy è marcata come «prosegui».

2.12 Validazione dei messaggi

Per attivare la validazione dei messaggi in transito sul gateway si accede al collegamento presente nella colonna *Validazione* presente tra gli elementi di configurazione della specifica erogazione/fruizione.

Compilare il form di configurazione (Fig. 2.97):

- *Stato*: Consente di abilitare/disabilitare la funzionalità di validazione sulla voce di configurazione scelta. L'opzione *warnignOnly* consente di attivare la funzionalità di validazione evitando però che, se tale fase non viene superata, venga bloccato il messaggio e restituito un errore. In quest'ultimo caso, gli errori di validazione verranno segnalati solo tramite l'emissione di opportuni messaggi diagnostici dal servizio di tracciamento.

Fig. 2.97: Validazione dei messaggi

- *Tipo*: Nel caso si sia abilitato il servizio di validazione, questo campo consente di selezionare la metodologia che si vuole utilizzare. I valori selezionabili da questo elenco cambiano in base alla tipologia delle API cui fa riferimento l'erogazione/fruizione.

I tipi di validazione previsti sono:

- *Wsdl 1.1*, la validazione si basa sull'interfaccia wsdl fornita con la API. Questo tipo di validazione è più rigorosa in quanto controlla non solo la conformità sintattica ma viene validato il messaggio in transito verificando che sia idoneo al PortType e Operation in uso. Questo tipo di validazione è applicabile solo al caso Soap.
- *Swagger 2.0 o OpenAPI 3.0*, nei casi in cui si è fornito un descrittore formale per una API Rest, la validazione sarà effettuata utilizzando gli strumenti associati allo specifico formato.
- *Schemi XSD*, la validazione si basa sugli schemi xsd allegati alle API. Utilizzato per la validazione sintattica dei messaggi XML sia nel caso Soap che Rest.

Nel caso di servizi Soap, se i messaggi che transitanto sulla porta di dominio possiedono il formato MTOM, per poterli validare è necessario attivare l'opzione *Accetta MTOM*. Tale opzione normalizza i messaggi prima di effettuarne la validazione e ripristina il formato originale una volta completato il processo di validazione.

Nota: Per la validazione dei messaggi riguardanti API REST con specifiche di interfaccia OpenAPI 3.x, è possibile attuare una configurazione avanzata del tipo di validazione effettuato. Maggiori dettagli vengono forniti nella sezione *Validazione dei messaggi con OpenAPI 3.x*.

2.12.1 Gestione differente tra Richiesta e Risposta

Per default, il tipo di validazione dei messaggi impostato riguarderà sia le richieste che le risposte.

È possibile differenziare il tipo di validazione registrando le seguenti *Proprietà* sull'erogazione o sulla fruizione:

- *validation.request.enabled* o *validation.response.enabled* : consentono di modificare l'impostazione configurata rispettivamente per la richiesta o la risposta. I valori associabili alle proprietà sono "true", "false" o "warning".
- *validation.request.type* o *validation.response.type* : consentono di modificare il tipo di validazione. Per una validazione basata sulla Specifica dell'API utilizzare il valore "interface", mentre per utilizzare solamente gli schemi indicare il valore "xsd".

- *validation.request.acceptMtom* o *validation.response.acceptMtom* : consentono di modificare l'impostazione configurata per i messaggi che possiedono il formato MTOM. I valori associabili alle proprietà sono "true" o "false".

2.12.2 Configurazione per API SOAP

È possibile configurare il tipo di validazione attuata su API SOAP registrando le seguenti *Proprietà* sull'erogazione o sulla fruizione:

- *validation.soapAction.enabled* : consente di disabilitare la verifica della SOAPAction. I valori associabili alle proprietà sono "true" o "false".

2.12.3 Configurazione per API REST

È possibile configurare il tipo di validazione attuata su API REST registrando le seguenti *Proprietà* sull'erogazione o sulla fruizione:

Nota: Tutte le proprietà configurate vengono verificate in AND tra di loro. Ad esempio è quindi possibile definire sia il codice http che il Content-Type per cui si desidera abilitare una validazione.

- *validation.emptyResponse.enabled* : consente di disabilitare la validazione della risposta in caso di payload http vuoto. I valori associabili alle proprietà sono "true" o "false". Per default questo controllo è abilitato.
- *validation.problemDetails.enabled* : consente di disabilitare la validazione della risposta nel caso il payload http contenga un oggetto *Problem Details* come definito nella specifica *RFC 7807* (https://tools.ietf.org/html/rfc7807). I valori associabili alle proprietà sono "true" o "false". Per default questo controllo è abilitato.
- *validation.returnCode* : consente di indicare i soli codici http per cui la validazione della risposta verrà effettuata. Possono essere associati differenti valori separati con la virgola, e ogni valore puà essere un codice o un intervallo di codice (es. 200-299,404). Per default viene verificato qualsiasi codice http.
- *validation.returnCode.not* : consente di impostare una validazione della risposta solamente per i messaggi che non corrispondono ai codici http definiti nella proprietà "validation.returnCode".
- *validation.contentType* : consente di indicare i soli Content-Type per cui la validazione della risposta verrà effettuata. Possono essere associati differenti Content-Type separati con la virgola, e possono essere utilizzati anche i tipi speciali "<type>/*" e "*/*" (es. text/xml,application/*). Per default viene verificato qualsiasi Content-Type.
- *validation.contentType.not* : consente di impostare una validazione della risposta solamente per i messaggi che non corrispondono ai Content-Type definiti nella proprietà "validation.contentType".

Nota: Per la validazione dei messaggi con specifiche di interfaccia OpenAPI 3.x, è possibile attuare una configurazione avanzata del tipo di validazione effettuato. Maggiori dettagli vengono forniti nella sezione *Validazione dei messaggi con OpenAPI 3.x*.

2.12.4 Opzioni Avanzate

È possibile modificare l'engine di validazione registrando le seguenti *Proprietà* sull'erogazione o sulla fruizione:

- *validation.buffer.enabled*: consente di abilitare o disabilitare il buffer che preserva i dati letti dallo stream. Se l'opzione viene disabilitata, il contenuto inoltrato al backend verrà ottenuto serializzando l'oggetto costruito in seguito alla lettura dello stream (es. serializzazione dell'elemento DOM in xml). I valori associabili alle proprietà sono "true" o "false".

2.13 Caching Risposta

In GovWay è possibile abilitare il salvataggio delle risposte in una cache sia globalmente, in modo che sia attivo per tutte le APIs, che singolarmente sulla singola erogazione o fruizione. Questa funzionalità permette ad un backend server di non dover riprocessare le stesse richieste più volte.

Tramite il collegamento *Caching Risposta*, presente nella sezione di configurazione della specifica erogazione/fruizione, è possibile agire sulla configurazione di tale funzionalità. L'impostazione permette di ridefinire la configurazione globale; i campi del form sono i medesimi descritti nella configurazione globale (sezione *Caching Risposta*).

2.14 Sicurezza a livello del messaggio

Tramite il collegamento *Sicurezza Messaggio*, presente nella sezione di configurazione della specifica erogazione/fruizione, è possibile impostare criteri di elaborazione dei messaggi in transito, attuati dal gateway, al fine di gestire i meccanismi di sicurezza previsti a livello del messaggio.

Il form presenta inizialmente lo *Stato* disabilitato. Per abilitare la sicurezza, impostare il valore dello stato su abilitato e confermare con il pulsante *Invia*. Appariranno gli elementi *Richiesta* e *Risposta*, come nella figura seguente.

Il form consente di selezionare uno schema di sicurezza, tra quelli disponibili, da applicare al messaggio di richiesta ed a quello di risposta. Gli schemi di sicurezza applicabili cambiano in base alla tipologia del messaggio sul quale si applica.

Per la gestione della sicurezza sul messaggio di richiesta, nel caso di una erogazione, il gateway agisce con il ruolo *Receiver* che comporta la seguente casistica:

- *Nel caso del protocollo SOAP*:
  - *WSSec Signature*, in ricezione si attende un messaggio firmato; l'azione è quella di verificare la firma presente
  - *WSSec Decrypt*, il messaggio ricevuto verrà decifrato
  - *WSSec SAML Token*, si attende un messaggio contenente una asserzione SAML; viene effettuata la verifica dell'asserzione presente.
  - *WSSec Username Token*, viene effettuata la validazione del token di autenticazione
  - *WSSec Timestamp*, se è prevista una scadenza all'interno del timestamp presente nel messaggio, se ne verificherà la validità
- *Nel caso del protocollo REST*
  - *JWT Decrypt*: il messaggio JSON ricevuto viene decifrato.
  - *JWT Verifier Signature*: al messaggio JSON ricevuto viene verificata la firma.

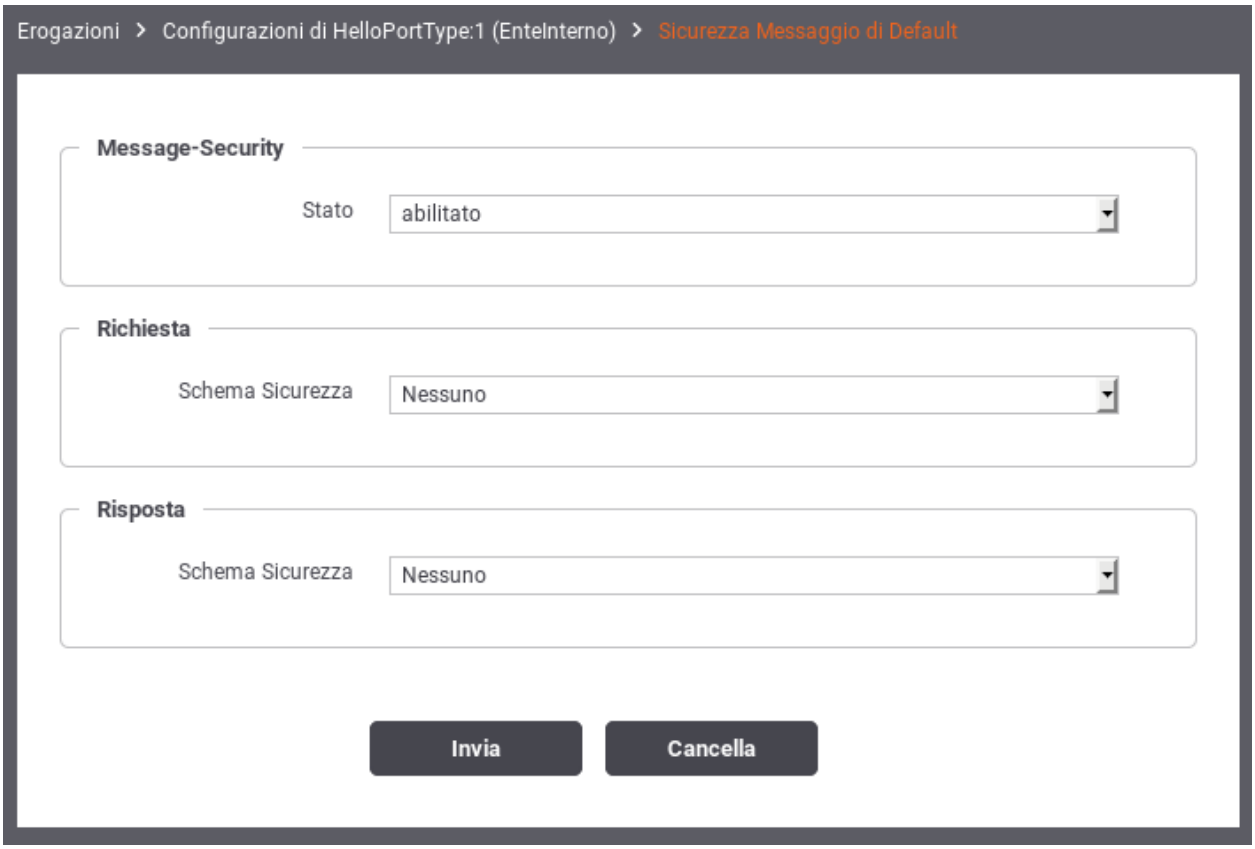


Fig. 2.98: Abilitazione Sicurezza Messaggio

  - *XML Decrypt*: il messaggio XML ricevuto viene decifrato.
  - *XML Verifier Signature*: al messaggio XML ricevuto viene verificata la firma.

Per la gestione della sicurezza sul messaggio di risposta, nel caso di una erogazione, il gateway agisce con il ruolo *Sender* che comporta la seguente casistica:

- *Nel caso del protocollo SOAP*:
  - *WSSec Signature*, il messaggio verrà firmato
  - *WSSec Encrypt*, il messaggio verrà cifrato
  - *WSSec SAML Token*, sul messaggio verrà inserita una asserzione SAML
  - *WSSec Username Token*, il messaggio verrà arrichito di un token di autenticazione
  - *WSSec Timestamp*, il messaggio verrà arrichito di una informazione temporale (tipicamente utilizzato insieme alla firma del messaggio)
- *Nel caso del protocollo REST*:
  - *JWT Encrypt*: il messaggio JSON di risposta viene cifrato prima dell'invio.
  - *JWT Signature*: il messaggio JSON di risposta viene firmato prima dell'invio.
  - *XML Encrypt*: il messaggio XML di risposta viene cifrato prima dell'invio.
  - *XML Signature*: il messaggio XML di risposta viene firmato prima dell'invio.

Nota: Si tenga presente che, nel caso di una fruizione, il ruolo del gateway si inverte diventando *Sender* nel caso della richiesta e *Receiver* nel caso della risposta. Gli schemi di sicurezza disponibili, nel caso della fruizione, rimangono quelli già descritti per Sender e Receiver.

2.15 Trasformazioni

Tra le attività di elaborazione, svolte dal gateway sui flussi di comunicazione in ingresso e uscita, vi è la possibilità di applicare delle *Regole di Trasformazione* che consentono di modificare dinamicamente i contenuti in transito prima che vengano instradati alla relativa destinazione.

2.15.1 Valori dinamici

Le regole di trasformazione possono avvalersi di un contesto di risorse, con valori aggiornati dinamicamente dal gateway, cui attingere per le trasformazioni da attuare. Tali risorse sono utilizzabili quando si procede con la definizione di una regola di trasformazione. Elenchiamo le risorse disponibili:

- *header:NAME* : valore dell'header http, corrispondente all'identificativo NAME, della richiesta.
- *query:NAME* : valore di un parametro della url di invocazione, corrispondente all'identificativo NAME.
- *form:NAME* : valore di un parametro della form, corrispondente all'identificativo NAME.
- *urlRegExp:EXPR* : applicazione di un'espressione regolare, rappresentata dal valore EXPR, alla url di invocazione (l'espressione deve avere un match con l'intera url).
- *xPath:EXPR* : applicazione di un'espressione XPath, rappresentata dal valore EXPR, alla richiesta xml (o soap).
- *jsonPath:EXPR* : applicazione di un'espressione jsonPath, rappresentata dal valore EXPR, alla richiesta json.
- *transaction:id* : l'identificativo UUID della transazione corrente.

- *date:FORMAT* : la data di elaborazione del messaggio; il formato fornito deve essere conforme a quanto richiesto dalla classe java "java.text.SimpleDateFormat" (es. ${date:yyyyMMdd_HHmmssSSS})
- *busta:FIELD* : accesso alle informazioni proprie del profilo di interoperabilità utilizzato; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.protocol.sdk.Busta" (ad es. per il mittente usare *busta.mittente*)
- *property:NAME*: accesso alle proprietà contenute nella traccia (ad esempio l'identificativo SDI); Il valore "NAME" indica il nome della proprietà da utilizzare.
- *tokenInfo:FIELD* : accesso ai claim di un token precedentemente validato; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.pdd.core.token.InformazioniToken" (es. per ottenere il valore del claim "sub" usare ${tokenInfo:sub})
- *aa:FIELD* : consente di accedere agli attributi recuperati tramite Attribute Authority; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.pdd.core.token.attribute_authority.InformazioniAttributi" (es. per ottenere il valore dell'attributo "attr1" usare ${aa:attributes[attr1]}, se configurata solamente 1 A.A., altrimenti usare ${aa:attributes[nomeAttributeAuthority][attr1]})
- *transportContext:FIELD* : accesso ai dati della richiesta http; il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe "org.openspcoop2.utils.transport.http.HttpServletTransportRequestContext" (es. per il principal usare ${transportContext:credential.principal})
- *config:NAME* : accesso alle proprietà configurate per l'API; il valore "NAME" indica la proprietà desiderata
- *clientApplicationConfig:NAME* : accesso alle proprietà configurate nell'applicativo fruitore; il valore "NAME" indica la proprietà desiderata
- *clientOrganizationConfig:NAME* : accesso alle proprietà configurate nel soggetto fruitore; il valore "NAME" indica la proprietà desiderata
- *providerOrganizationConfig:NAME* : accesso alle proprietà configurate nel soggetto erogatore; il valore "NAME" indica la proprietà desiderata
- *system:NAME* : valore associato alla proprietà di sistema, indicata nella configurazione generale, con nome "NAME"
- *env:NAME* : valore associato alla variabile di sistema con nome "NAME"
- *java:NAME* : valore associato alla variabile java con nome "NAME"

Per le risposte sono inoltre disponibili anche le seguenti risorse:

- headerResponse.NAME: valore dell'header http, corrispondente all'identificativo NAME, della risposta.
- xPathResponse.EXPR: applicazione di un'espressione XPath, rappresentata dal valore EXPR, alla risposta xml (o soap).
- jsonPathResponse.EXPR: applicazione di un'espressione jsonPath, rappresentata dal valore EXPR, alla risposta json.
- dateResponse.FORMAT: la data di elaborazione della risposta; il formato fornito deve essere conforme a quanto richiesto dalla classe java "java.text.SimpleDateFormat" (es. ${date:yyyyMMdd_HHmmssSSS})

L'utilizzo dei suddetti elementi, come placeholder all'interno di template, comporta l'automatica sostituzione con il valore attuale a runtime da parte del gateway.

La sintassi per accedere le proprietà dinamiche sopraelencate è differente in base allo specifico contesto di utilizzo. Se si tratta di un testo interpretato direttamente da GovWay le proprietà saranno direttamente accessibili utilizzando il seguente formato:

- ${header:NAME} o ${headerResponse:NAME}

- ${query:NAME}
- ${form:NAME}
- ${xPath:EXPR} o ${xPathResponse:EXPR}
- ${jsonPath:EXPR} o ${jsonPathResponse:EXPR}
- ${urlRegExp:EXPR}
- ${transaction:id}
- ${date:FORMAT} o ${dateResponse:FORMAT}
- ${busta:FIELD}
- ${property:NAME}
- ${tokenInfo:FIELD}
- ${aa:FIELD}
- ${transportContext:FIELD}
- ${config:NAME}
- ${clientApplicationConfig:NAME}
- ${clientOrganizationConfig:NAME}
- ${providerOrganizationConfig:NAME}
- ${system:NAME}
- ${env:NAME}
- ${java:NAME}

Nei casi in cui il testo della trasformazione è interpretato da framework esterni (quali Freemarker o Velocity) le proprietà vengono rese disponibili da Govway inizializzando una mappa contenente i valori come oggetti. In questo caso le chiavi della mappa sono le seguenti (tra parentesi sono indicati i tipi di dato corrispondenti):

- header o headerResponse (java.util.Map<String, String>); in caso di molteplici header con stesso nome è disponibile la variabile headerValues o headerResponseValues (java.util.Map<String, List<String>>)
- query (java.util.Map<String, String>); in caso di molteplici parametri con stesso nome è disponibile la variabile queryValues (java.util.Map<String, List<String>>)
- form (java.util.Map<String, String>); in caso di molteplici parametri con stesso nome è disponibile la variabile formValues (java.util.Map<String, List<String>>)
- xPath o xPathResponse (org.openspcoop2.pdd.core.dynamic.PatternExtractor)
- jsonPath o jsonPathResponse (org.openspcoop2.pdd.core.dynamic.PatternExtractor)
- urlRegExp (org.openspcoop2.pdd.core.dynamic.URLRegExpExtractor)
- transactionId (java.lang.String)
- date (java.util.Date)
- busta (org.openspcoop2.protocol.sdk.Busta)
- property (java.util.Map<String, String>)
- tokenInfo (org.openspcoop2.pdd.core.token.InformazioniToken)
- aa (org.openspcoop2.pdd.core.token.attribute_authority.InformazioniAttributi)
- transportContext (org.openspcoop2.utils.transport.http.HttpServletTransportRequestContext)

- config (java.util.Map<String, String>)
- clientApplicationConfig (java.util.Map<String, String>)
- clientOrganizationConfig (java.util.Map<String, String>)
- providerOrganizationConfig (java.util.Map<String, String>)
- system (org.openspcoop2.pdd.core.dynamic.PropertiesReader)
- env (org.openspcoop2.pdd.core.dynamic.PropertiesReader)
- java (org.openspcoop2.pdd.core.dynamic.PropertiesReader)

Nel caso di utilizzo di template "Freemarker" o "Velocity" sono disponibili i seguenti ulteriori oggetti:

- class; permette di definire classi. L'utilizzo varia a seconda del tipo di template engine:
  - velocity: class.forName(«my.package.name»)
  - freemarker: class[«my.package.name»]
- new; permette di istanziare una classe. L'utilizzo varia a seconda del tipo di template engine:
  - velocity: new.instance(«my.package.name»,»Parametro1»,»ParametroN»)
  - freemarker: new(«my.package.name»,»Parametro1»,»ParametroN»)
- transportContext (org.openspcoop2.utils.transport.http.HttpServletTransportRequestContext); permette di accedere ai dati della richiesta http (servlet request, principal . . .)
- request/response: permette di accedere al contenuto della richiesta/risposta (org.openspcoop2.pdd.core.dynamic.ContentExtractor)
- attachments (org.openspcoop2.pdd.core.dynamic.AttachmentsReader); consente di ottenere gli allegati registrati sull'API
- context (java.util.Map<String, Object>); permette di accedere al contesto della richiesta.
- errorHandler (org.openspcoop2.pdd.core.dynamic.ErrorHandler); permette di generare risposte personalizzate che segnalano l'impossibilità di proseguire la trasformazione.

Nel caso di utilizzo di template "ZIP", "TGZ" o "TAR" sono disponibili le seguenti le proprietà dinamiche, interpretate direttamente da GovWay, utilizzabili per accedere a parti della richiesta o della risposta:

- ${content} : payload http del messaggio
- ${soapEnvelope} : soap envelope del messaggio
- ${soapBody} : contenuto del soap body
- ${attachment[index]} : attachment presente in un messaggio multipart alla posizione indicata dall'intero "index"
- ${attachmentId[id]} : attachment presente in un messaggio multipart che possiede il Content-ID indicato

2.15.2 Trasformazione

Nel contesto della configurazione specifica di una erogazione o di una fruizione si può accedere alla funzionalità «Trasformazioni» per inserire una lista di definizioni che applicano trasformazioni ai flussi in entrata e/o uscita. Le trasformazioni create hanno la struttura di una lista ordinata e a ciascun elemento della lista è associato un insieme di criteri di applicabilità. La logica del gateway è quella di analizzare le trasformazioni nell'ordine della lista, selezionando la prima di esse i cui criteri di applicabilità sono tutti soddisfatti.

Tramite il pulsante *Aggiungi* è possibile inserire una nuova trasformazione (Fig. 2.99).

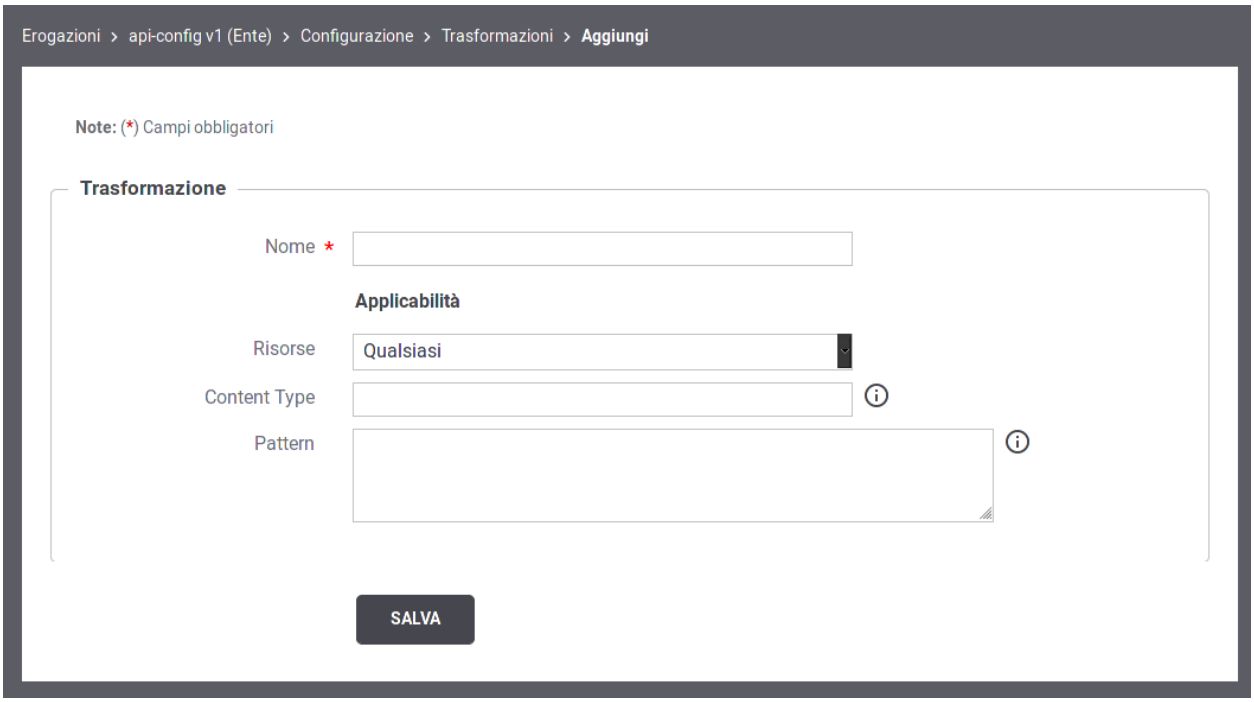


Fig. 2.99: Nuova Trasformazione

La creazione di una trasformazione richiede che vengano inseriti i seguenti dati:

- Nome: identificativo che rappresenta il nome assegnato alla trasformazione
- Applicabilità: sono i campi che vanno a comporre il criterio di applicabilità della trasformazione:
  - Risorse/Azioni: le operazioni sulle quali è applicabile la trasformazione.
  - Content-Type: i content-type sui quali è applicabile la trasformazione.
  - Pattern: il pattern inserito viene confrontato con il messaggio di richiesta del flusso di comunicazione al fine di verificare l'eventuale match. Il pattern può essere espresso nella sintassi «XPath», nel caso di messaggi XML, o JSONPath, nel caso di messaggi JSON.

Le trasformazioni create sono visualizzate nella forma di elenco ordinato (Fig. 2.100). L'icona iniziale di ciascun elemento consente di modificarne la posizione.

Ciascuna regola elencata visualizza i dati che sono stati forniti come criterio di applicabilità. A quelli inseriti in fase di creazione si aggiungono i Soggetti e gli Applicativi, che possono essere forniti accedendo i rispettivi collegamenti. I soggetti/applicativi associati ad una regola saranno confrontati con l'identità del soggetto/applicativo mittente di ciascuna richiesta.

Accedendo il dettaglio di una regola di trasformazione vengono presentate le due sezioni:

- Trasformazione: per aggiornare il nome o i criteri di applicabilità.
- Regole di Trasformazione: per aggiornare le regole di trasformazione attuate sulla richiesta e sulla risposta.

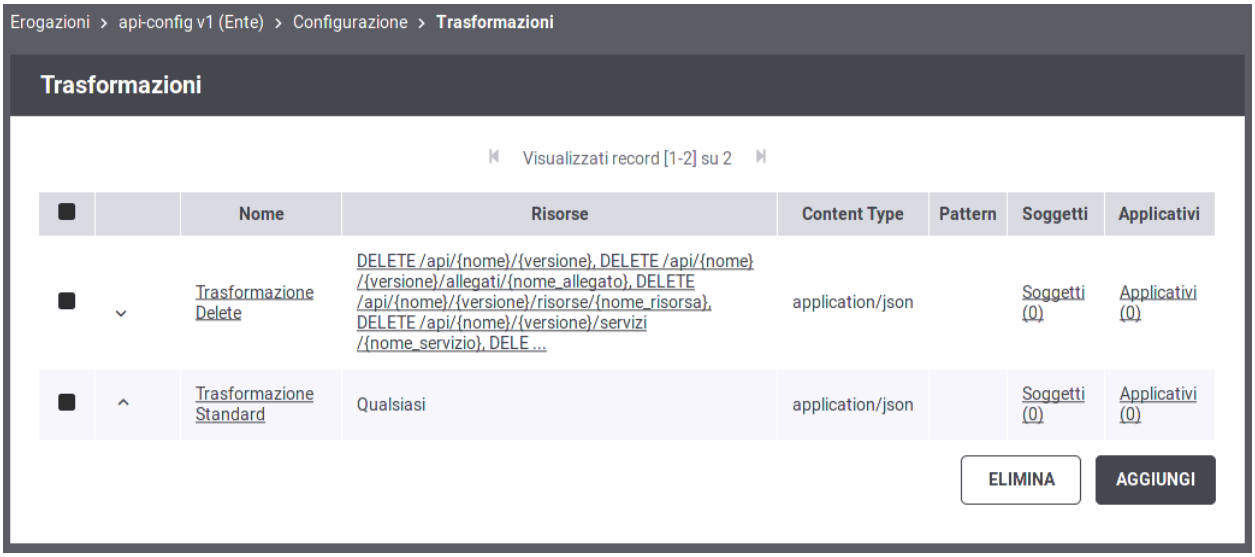


Fig. 2.100: Lista regole di trasformazione

Regole di Trasformazione della Richiesta

Selezionando il collegamento «Richiesta», nel riquadro delle Regole di Trasformazione, si procede con la definizione formale della trasformazione attuata sulle richieste in ingresso sulle quali è applicabile la trasformazione corrente. Le trasformazioni possono essere applicate sia a livello del trasporto che del contenuto, come mostrano le sezioni visualizzate in Fig. 2.101.

Fig. 2.101: Regola di trasformazione della richiesta

A livello del trasporto è possibile applicare trasformazioni sugli «HTTP Headers», selezionando l'omonimo collegamento e quindi aggiungendo le operazioni da effettuare (Fig. 2.102).

Fig. 2.102: Operazioni sugli Header HTTP

Ciascuna operazione può essere selezionata tra le seguenti:

- add: per aggiungere un nuovo header specificando successivamente nome e valore
- delete: per eliminare un header indicandone successivamente il nome
- update: per modificare un header indicandone successivamente il nome ed il nuovo valore
- updateOrAdd: per modificare un header indicandone successivamente il nome ed il nuovo valore. Nel caso l'header non si presente, verrà aggiunto.

Nota: i valori specificati per gli header http possono contenere le proprietà dinamiche descritte nella sezione *Valori dinamici*. Il campo "Identificazione Fallita" permette di definire il comportamento del Gateway quando non riesce a risolvere parti dinamiche contenute nel valore indicato. Le configurazioni utilizzabili sono:

- Termina con errore: la transazione termina con un errore che riporta la fallita risoluzione della parte dinamica indicata per il valore;
- Continua senza header: la transazione continua senza completare la gestione dell'header.

Sempre a livello del trasporto è possibile applicare trasformazioni anche sui parametri presenti nella Query String, selezionando il collegamento «URL Parameters». La modalità di configurazione è del tuto analoga a quanto appena descritto per gli Header HTTP.

Abilitando l'opzione sul Contenuto è possibile procedere con la definizione di operazioni sul contenuto della richiesta (Fig. 2.103).

Per la modifica del contenuto della richiesta devono essere forniti i seguenti dati:

- Tipo Conversione: indica il tipo di trasformazione da applicare al contenuto. Si può scegliere una tra le seguenti opzioni:
  - HTTP Payload Vuoto: opzione presente nel caso REST. Il contenuto della richiesta diventa un payload http vuoto.

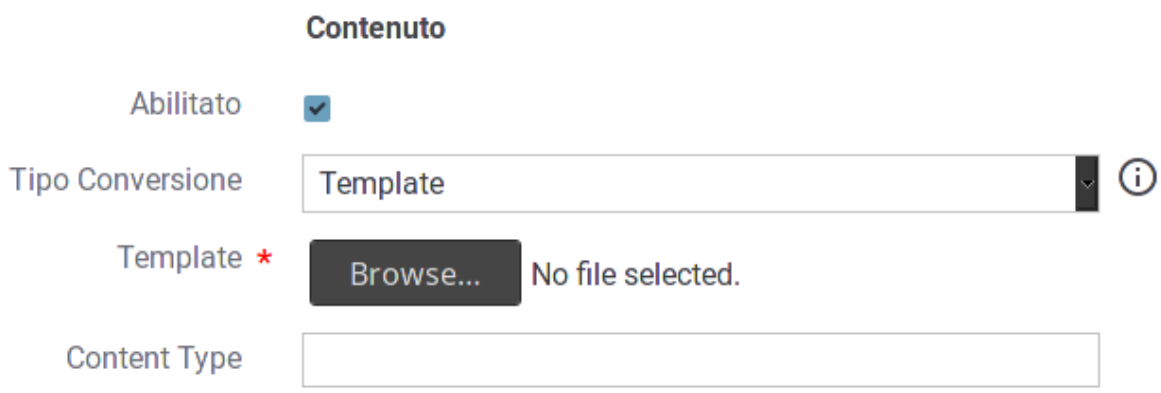


Fig. 2.103: Modifica del Contenuto della Richiesta

  - SOAP Body Vuoto: opzione presente nel caso SOAP. Il contenuto della richiesta diventa un messaggio SOAP con SoapBody vuoto.
  - Template: il contenuto della richiesta viene assegnato utilizzando il template fornito in configurazione, che può contenere parti dinamiche definite tramite una sintassi proprietaria di GovWay.
  - Freemarker Template: il contenuto della richiesta viene assegnato utilizzando il template «Freemarker» (https://freemarker.apache.org/) fornito in configurazione.
  - Freemarker Template (Archivio Zip): il file fornito deve essere un archivio zip contenenti dei files che rispettano la sintassi del template engine "Freemarker". Viene richiesta la presenza, all'interno dell'archivio zip, di un file indice che possieda il nome "index.ftl".
  - Velocity Template: il contenuto della richiesta viene assegnato utilizzando il template «Velocity» (http://velocity.apache.org/) fornito in configurazione.
  - Velocity Template (Archivio Zip): il file fornito deve essere un archivio zip contenenti dei files che rispettano la sintassi del template engine "Velocity". Viene richiesta la presenza, all'interno dell'archivio zip, di un file indice che possieda il nome "index.vm".
  - XSLT: il contenuto della richiesta viene modificato applicando la trasformazione XSLT fornita in configurazione. Questo metodo è applicbile nel caso di messaggi XML o SOAP.
  - ZIP Compressor: il contenuto della richiesta verrà trasformato in un archizio zip il cui contenuto viene definito dal file fornito che deve contenere proprietà indicate come nome=valore in ogni linea. Il nome della proprietà corrisponde all'entry name all'interno dell'archivio (es. dir/subDir/entryName1). Il valore della proprietà corrisponde al contenuto dell'entry. È possibile selezionare parti del messaggio, per associarle come contenuto dell'entry, utilizzando le espressioni dinamiche risolte a runtime dal Gateway (sezione *Valori dinamici*).
  - TGZ Compressor: il contenuto della richiesta verrà trasformato in un archizio tgz il cui contenuto è definito tramite il file fornito che deve possedere la medesima struttura descritta per il tipo "ZIP".
  - TAR Compressor: il contenuto della richiesta verrà trasformato in un archizio tar il cui contenuto è definito tramite il file fornito che deve possedere la medesima struttura descritta per il tipo "ZIP".
- Template: nei casi che lo prevedono, con questo elemento si fornisce il template da utilizzare per ottenere il nuovo contenuto della richiesta.
- Content-Type: opzionalmente, tramite questo elemento, è possibile assegnare un content-type alla richiesta modificata.

Nota: i template possono contenere le proprietà dinamiche descritte nella sezione *Valori dinamici*. La sintassi adottata dipende dal template. Una finestra di help contestuale, presente nell'interfaccia, guiderà l'utente nell'applicazione della sintassi corretta.

Conversione da REST a SOAP

Una particolare trasformazione del contenuto della richiesta è quella di convertire il formato da REST a SOAP. Questa funzionalità si ottiene abilitando la sezione «Trasformazione SOAP», presente nel caso di servizi REST. I dati da fornire per la configurazione sono (Fig. 2.104):

- Versione: selezione della versione del protocollo SOAP
- SOAP Action: indicazione della SOAP Action da utilizzare
- Imbustamento SOAP: se il messaggio ottenuto con le operazioni di trasformazione applicate non è in formato SOAP è possibile decidere di far generare al gateway gli elementi di imbustamento. Le opzioni possibili sono:
  - Disabilitato: nessun imbustamento.
  - Utilizza contenuto come SOAP Body: il contenuto attuale viene utilizzato come SOAP Body nel contesto dell'envelope creato.
  - Utilizza contenuto come Attachment: il contenuto attuale viene inserito come attachment relativo al messaggio SOAP generato. Se viene selezionata questa opzione dovranno essere forniti ulteriori dati, quali:
    - Content Type Attachment: è possibile specificare un Content-Type per l'attachment.
    - SOAP Body: stabilire quale deve essere il contenuto del SOAP Body. Per questo punto si procede analogamente a quanto già descritto per la trasformazione del contenuto principale della richiesta.

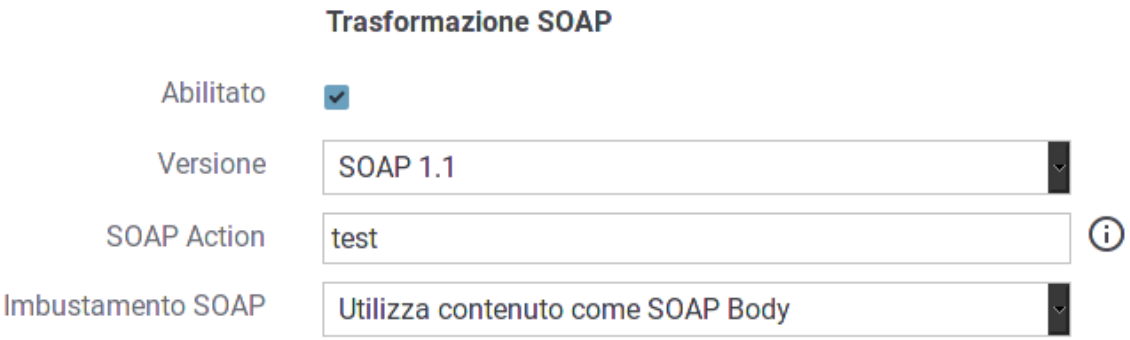


Fig. 2.104: Conversione da REST a SOAP

Conversione da SOAP a REST

Una particolare trasformazione del contenuto della richiesta è quella di convertire il formato da SOAP a REST. Questa funzionalità si ottiene abilitando la sezione «Trasformazione REST», presente nel caso di servizi SOAP. I dati da fornire per la configurazione sono (Fig. 2.105):

- Path: path della risorsa cui deve fare riferimento il nuovo messaggio di richiesta REST-
- HTTP Method: il metodo HTTP utilizzato.

Fig. 2.105: Conversione da SOAP a REST

Regole di Trasformazione della Risposta

Analogamente a quanto visto per la richiesta è possibile utilizzare il link «Risposte», nell'area «Regole di Trasformazione», per procedere con l'impostazione di regole per trasformare le risposte. A differenza del caso della richiesta, dove si può definire un unico meccanismo di trasformazione, in questo caso è possibile definire diverse regole di trasformazione basate sulla casistica delle risposte che si può presentare.

Quando si aggiunge una nuova regola di trasformazione della risposta si procede inserendo le seguenti informazioni (Fig. 2.106):

- Nome: nome assegnato alla regola di trasformazione
- Codice Risposta: Come criterio di applicabilità della regola, è possibile indicare il codice di risposta con le seguenti opzioni:
  - Qualsiasi: qualunque codice di risposta ottenuto
  - Singolo: si inserisce unp specifico codice di risposta per il quale è applicabile la regola
  - Intervallo: si inseriscono gli estremi dell'intervallo di codici di risposta per il quale è applicabile la regola
- Content-Type: criterio di corrispondenza con uno dei content-type indicati
- Pattern: espressione XPath o JsonPath da confrontare con il contenuto della risposta per un eventuale match

Le operazioni di trasformazione sulla risposta sono attuabili in maniera del tutto analoga a quanto già descritto per la richiesta. Diversamente dal caso della richiesta, al posto delle modifiche sui parametri della URL (non presenti nella risposta) è possibile modificare il Codice Risposta restituito.

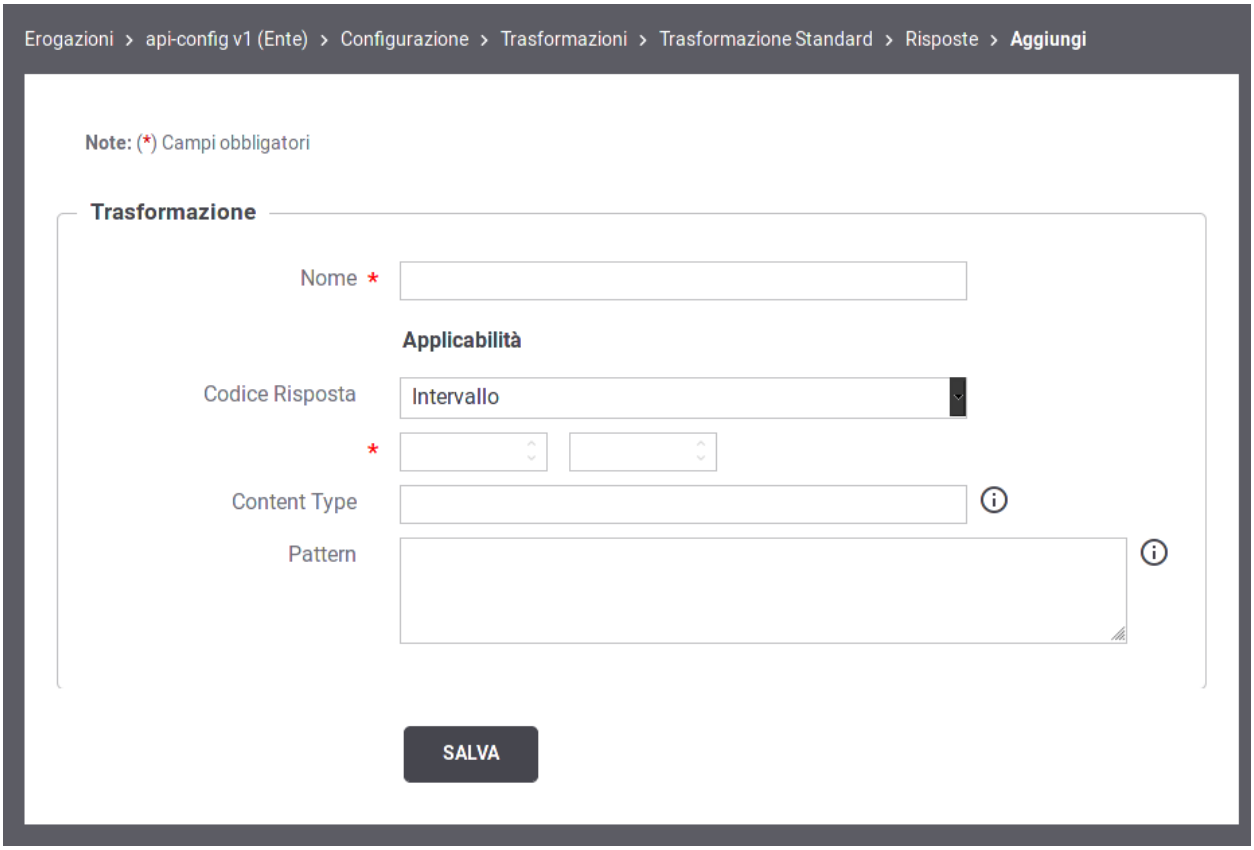


Fig. 2.106: Creazione regola di trasformazione della risposta

Nota: Se sulla richiesta si è scelto di attuare la conversione da SOAP a REST, o viceversa, la trasformazione complementare risulterà disponibile anche nella configurazione della risposta.

2.16 Tracciamento

Il tracciamento è la funzionalità del gateway che comporta la registrazione dei dati relativi alle comunicazioni in transito riguardanti i servizi erogati e fruiti. Nella logica del gateway, tutte le informazioni che riguardano una singola interlocuzione, a partire dalla richiesta pervenuta fino alla conclusione con l'invio dell'eventuale risposta, sono riconducibili ad un'unica entità denominata *Transazione*.

Una transazione registrata dal gateway ha la seguente struttura:

- *Dati di Identificazione Generale*. Sono le informazioni che identificano la comunicazione specifica in termini dei soggetti coinvolti e del servizio richiesto: Soggetto Erogatore, Soggetto Fruitore, Servizio, Azione, Esito, ...
- *Dati della Richiesta*. Sono le informazioni di dettaglio relative alla richiesta: Identificativo del Messaggio, Timestamp di ingresso, Timestamp di uscita, dimensioni del messaggio, ...
- *Dati della Risposta*. Sono le medesime informazioni già citate al punto precedente, ma relative alla comunicazione di risposta.
- *Traccia Richiesta*. La traccia emessa dal gateway con i dettagli relativi alla richiesta.
- *Traccia Risposta*. La traccia emessa dal gateway con i dettagli relativi alla risposta.
- *Messaggi Diagnostici*. La sequenza dei messaggi diagnostici, ordinati cronlogicamente, emessi dal gateway nel corso dell'elaborazione dell'intera transazione.
- *Fault di Ingresso*. Viene registrato come Fault di Ingresso l'eventuale messaggio di errore ricevuto dal gateway durante l'invocazione di un servizio (interno o esterno al dominio gestito).
- *Fault di Uscita*. Viene registrato come Fault di Uscita l'eventuale messaggio di errore inoltrato dal gateway al mittente della richiesta (interno o esterno al dominio gestito), dopo aver ricevuto un fault dal servizio invocato.
- *Parametri e Misurazioni*. Sono i parametri e le misurazioni che riguardano la transazione, come ad esempio: l'identificativo della transazione, le url invocate, i tempi di latenza, ...

In questa sezione è possibile personalizzare la configurazione di default del tracciamento definita in accordo a quanto descritto in sezione *Tracciamento*. Le personalizzazioni inserite in questo contesto avranno validità per le sole comunicazioni riguardanti la specifica erogazione/fruizione (Fig. 2.107).

Le sezioni presenti nella pagina sono:

- *Transazioni Registrate*: l'utente ha l'opzione per mantenere il default definito nella sezione di configurazione generale (sezione *Tracciamento*) oppure ridefinirlo.
- *Messaggi Diagnostici*: l'utente ha l'opzione per mantenere il default definito nella sezione di configurazione generale (sezione *Tracciamento*) oppure ridefinire il criterio per la sola memorizzazione su Database.
- *Correlazione Applicativa*: consente di impostare delle regole per estrarre dai messaggi in transito, codici, riferimenti, o altri contenuti al fine di arricchire i dati tracciamento generati dal gateway (sezione *Correlazione Applicativa*).

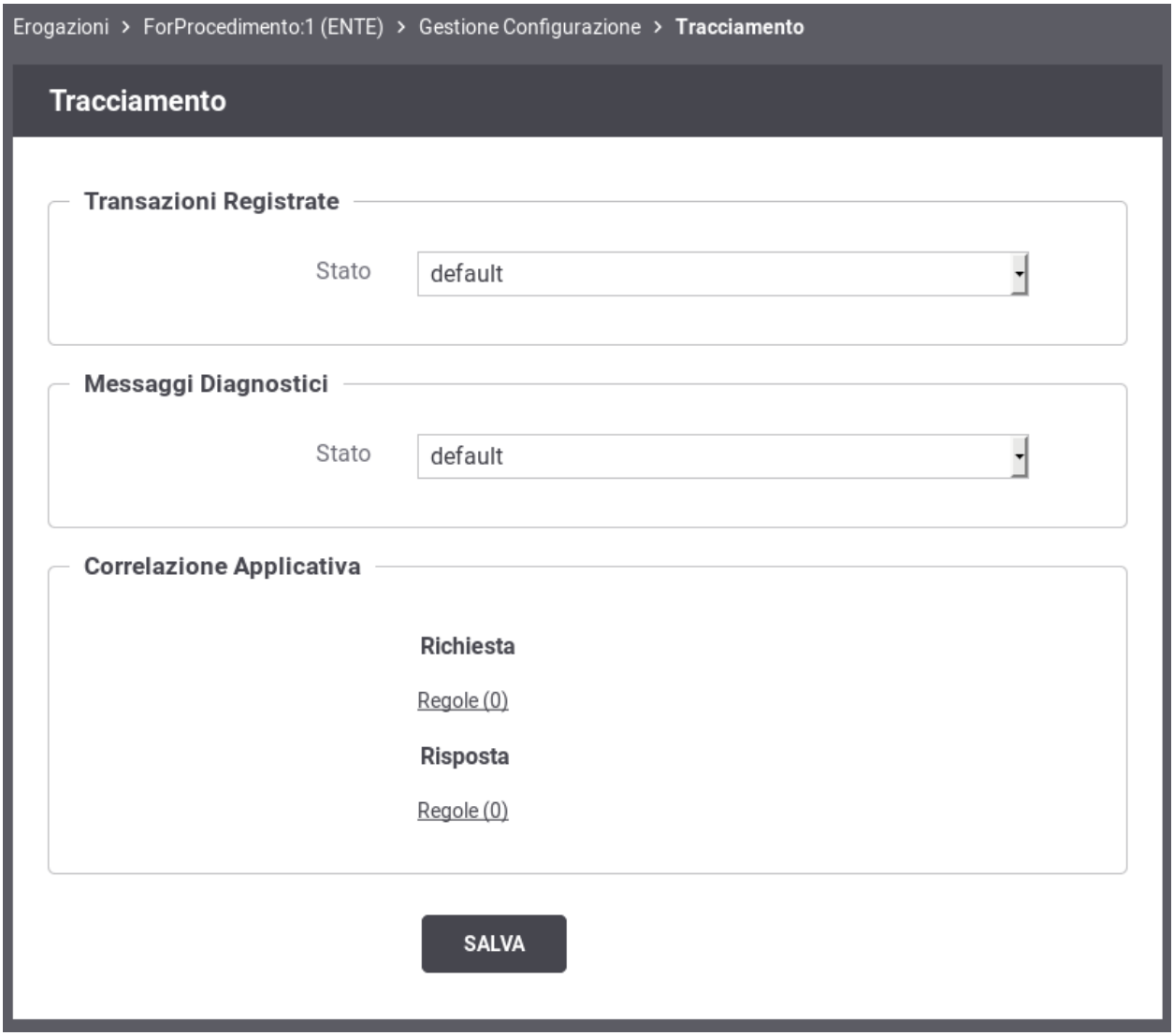


Fig. 2.107: Tracciamento per la singola erogazione/fruizione

2.17 Correlazione Applicativa

La funzione di *Correlazione Applicativa* consente al gateway che elabora il messaggio di richiesta, di estrarre un identificatore relativo al contenuto applicativo. L'identificatore, se presente, finisce nei sistemi di tracciamento e diagnostici, a completamento delle informazioni già presenti. I dati per configurare la correlazione applicativa consistono in un insieme di regole per l'estrazione di tale identificatore.

Per accedere alla configurazione della correlazione applicativa, per una data erogazione/fruizione, si utilizza la sezione «Correlazione Applicativa» presente nell'ambito della configurazione del tracciamento di una fruizione/erogazione (sezione *Tracciamento*).

Utilizzando il collegamento *Regole*, presente nel riquadro della Richiesta o Risposta, si accede all'elenco delle regole di correlazione applicativa presenti. Premere il pulsante *Aggiungi* per aggiungere una nuova regola (Fig. 2.108)

Fig. 2.108: Creazione di una regola di correlazione applicativa

Per la creazione di una regola di correlazione applicativa si devono indicare i seguenti dati:

- *Elemento*: Questo dato serve per capire su quali messaggi è applicabile la regola di correlazione applicativa che si sta definendo. Lasciando il campo vuoto si intende che la regola si applica a tutti i messaggi. In alternativa è possibile indicare:
  - *Nome Azione o Risorsa*: il nome esatto dell'azione o della risorsa su cui verrà applicativa la regola
  - *HttpMethod e Path* (utilizzabile solo su API REST): metodo http e path di una risorsa dell'API; è possibile indicare qualsiasi metodo o qualsiasi path con il carattere speciale "*". È inoltre possibile definire solamente la parte iniziale di un path attraverso lo "*". Alcuni esempi:
    - "POST /resource"
    - "* /resource"
    - "POST *"
    - "* /resource/*"

  - *XPath o JSONPath*: Espressione che può rappresentare un XPath o JSONPath. Se l'espressione ha un match con il contenuto la regola verrà applicata
  - *LocalName dell'elemento xml*: in caso il messaggio sia un xml (soap o rest), è possibile indicare il local name del root element xml su cui verrà applicativa la regola

- *Modalità Identificazione*: rappresenta la modalità di acquisizione dell'identificatore applicativo. Può assumere i seguenti valori:
  - *Url di Invocazione*: il valore viene preso dalla url utilizzata dal servizio applicativo per l'invocazione. La regola per l'estrazione dalla url viene specificata tramite un'espressione regolare inserita nel campo pattern (l'espressione deve avere un match con l'intera url).
  - *Contenuto*: Il valore viene estratto direttamente dal messaggio applicativo. La regola per l'estrazione dal messaggio è specificata tramite un'espressione XPath o JSONPath inserita nel campo pattern;
  - *Header HTTP*: Il valore viene estratto dall'header di trasporto avente il nome indicato nel campo successivo.
  - *Header di Integrazione*: il valore viene estratto dall'header di integrazione GovWay presente nel valore della proprietà *IDApplicativo*.
  - *Disabilitata*: l'identificatore applicativo non viene estratto. Questa opzione è utile quando si vuole disabilitare l'estrazione dell'id applicativo solo per specifici messaggi;
- *Pattern*: definisce l'espressione regolare, nel caso di identificazione urlBased, o l'espressione XPath/JSONPath, nel caso di identificazione contentBased, utilizzata per l'acquisizione dell'identificatore applicativo.
- *Identificazione Fallita*: azione da intraprendere nel caso fallisca l'estrazione dell'identificatore applicativo tramite la regola specificata. Nel caso sia stato indicato *blocca*, tali richieste non verranno accettate con restituzione di un errore al mittente;
- *Riuso ID*: opzione per abilitare/disabilitare il riuso dell'identificatore del messaggio (assegnato dal gateway) nel caso in cui vengano inviati messaggi con identificatori applicativi già processati in precedenza.

2.18 MTOM

Nei casi in cui il mittente e il destinatario si scambiano messaggi con allegati (nell'ambito del protocollo SOAP), utilizzando il protocollo MTOM, GovWay è in grado di gestire tali comunicazioni in modalità trasparente e quindi senza alcun intervento.

In altre situazioni è possibile sfruttare le funzionalità di GovWay per beneficiare delle ottimizzazioni del protocollo MTOM quando uno dei due interlocutori non è in grado di supportare tale protocollo, oppure per effettuare verifiche di congruità dei messaggi in transito basati su MTOM.

Nel caso di una erogazione, per il messaggio di richiesta, le opzioni disponibili sono:

- *disable*. Non viene svolta alcuna azione.
- *unpackaging*. In questo scenario il client fruitore invia dati binari nel formato MTOM ma l'erogatore non supporta tale formato. Il gateway effettua la trasformazione del messaggio inserendo i dati binari in modalità *Base64 encoded* prima che venga inviato al destinatario. Sulla risposta sarà effettuato il processo inverso.
- *verify*. Sia il fruitore che l'erogatore utilizzano MTOM ma si vogliono validare i messaggi. Il gateway effettua, tramite opportuni pattern xpath forniti, la validazione dei messaggi al fine di verificare la conformità del formato del messaggio rispetto a quanto atteso dall'erogatore.

Sempre nel caso di una erogazione, per il messaggio di risposta, le opzioni disponibili sono:

- *disable*. Non viene svolta alcuna azione.

- *packaging*. In questo scenario il client fruitore invia dati binari nella modalità Base64 encoded ma l'erogatore richiede il formato MTOM. Il gateway effettua la trasformazione del messaggio secondo il protocollo MTOM prima che venga inviato al destinatario. Sulla risposta sarà effettuato il processo inverso.
- *verify*. Analogo a quanto descritto per il messaggio di richiesta.

Nota: Nel caso si utilizzi la validazione dei contenuti, basata su xsd o wsdl, è possibile che la struttura MTOM non sia stata prevista negli schemi e quindi faccia fallire l'esito della stessa. In questo caso, quando si attiva la validazione è necessario abilitare l'opzione *Accetta MTOM/XOP-Message* affinché il processo di validazione tenga conto del formato MTOM.

Nota: Nel caso di una fruizione, le opzioni di configurazione disponibili per la richiesta diventano quelle per la risposta e viceversa.

2.19 Registrazione Messaggi

Nella sezione *Tracciamento* sono descritte le configurazioni per attivare il salvataggio dei messaggi in transito sul gateway. In questa sezione si ha la possibilità di ridefinire le opzioni di configurazione, stabilite a livello generale, al fine di personalizzare il servizio di registrazione dei messaggi per la specifica configurazione dell'erogazione/fruizione.

Per la descrizione delle opzioni di configurazione si faccia riferimento alla sezione generale precedentemente indicata.

2.20 Proprietà

Ad una API è possibile associare una serie di proprietà consultabili da una qualsiasi delle funzionalità precedentemente descritte.

(Fig. 2.19).

Questa funzionalità è frequentemente utilizzata in combinazione con le *Trasformazioni* per poter permettere all'utente di configurarne il comportamento senza dover modificare e caricare un nuovo file template di trasformazione. All'interno di un template di trasformazione è possibile accedere alle proprietà tramite la sintassi "config" come descritto nella sezione *Valori dinamici*.

Le proprietà permettono inoltre di effettuare la configurazione di aspetti avanzati di una funzionalità che non rientrano nel suo normale utilizzo. Ad esempio è possibile differenziare il comportamento della validazione dei messaggi, tra richiesta e risposta, utilizzando le proprietà descritte nella sezione *Validazione dei messaggi*.

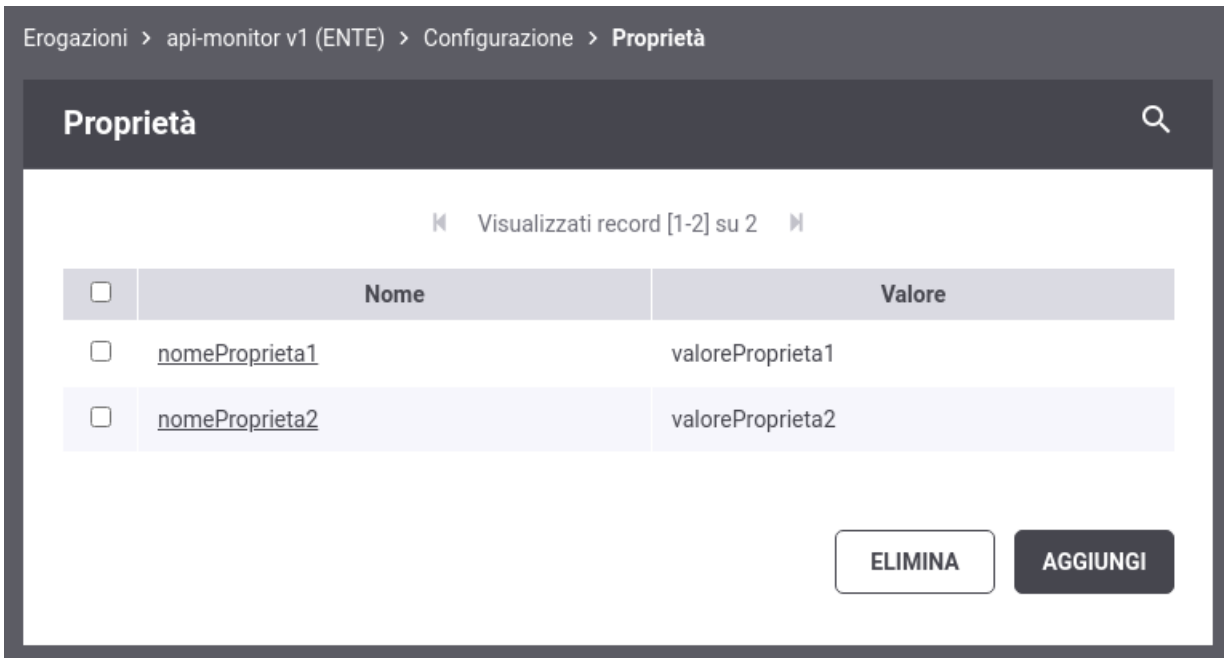


Fig. 2.109: Elenco di proprietà di una API

CAPITOLO 3

Profilo “ModI”

Il profilo “ModI” consente in maniera del tutto trasparente alle applicazioni interne al dominio, la conformità delle API (sia in fruzione che in erogazione) alle nuove *Linee Guida AGID di Interoperabilità* (https://www.agid.gov.it/it/infrastrutture/sistema-pubblico-connettivita/il-nuovo-modello-interoperabilita).

La struttura complessiva del processo di configurazione si mantiene analoga a quanto già descritto per il profilo API Gateway. Le differenze, con rispetto al profilo API Gateway, presentate in questa sezione, riguardano vincoli sulle scelte operabili dalla console e le informazioni di configurazione aggiuntive specifiche per la realizzazione degli scenari in accordo al Modello di Interoperabilità.

3.1 Concetti Preliminari

Il Modello di Interoperabilità mantiene sostanzialmente invariato il concetto di *dominio* di un’amministrazione rispetto a quanto prevedeva il precedente modello SPCoop. Resta quindi fondamentale individuare il perimetro d’azione delle interfacce dei servizi rispetto al sistema informativo dell’ente e i propri interlocutori.

Il concetto di dominio, che riveste particolare importanza nella gestione degli aspetti di sicurezza, si sposa perfettamente con i modelli di configurazione di GovWay dove è possibile attivare:

- *erogazioni di API*: richieste che provengono da amministrazioni esterne al dominio e sono dirette ad applicativi interni.
- *fruizioni di API*: richieste che provengono dagli applicativi interni del dominio e sono dirette verso altre amministrazioni.

La govwayConsole, all’atto della registrazione di Soggetti (Enti/Organizzazioni) e Applicativi (Sistemi/Applicazioni di un ente), consente di specificarne il *Dominio*, interno o esterno, al fine della corretta rappresentazione degli scenari di configurazione dei servizi.

Il profilo ModI prevede che i servizi siano basati su SOAP o REST fornendo sempre un descrittore formale delle interfacce basato su uno specifico IDL (Interface Description Language):

- WSDL 1.1 e successivi, per la descrizione delle interfacce SOAP
- OpenAPI 3.0 e successivi, per la descrizione delle interfacce REST

Nel processo di configurazione, tramite la govwayConsole, sono inoltre tenuti in considerazione tutti gli aspetti previsti dalle Linee Guida:

- *Pattern di Interazione*: definiscono la modalità con cui fruitore ed erogatore di un servizio interagiscono. Sono previsti i seguenti pattern:
  - *Bloccante*: il fruitore invia la richiesta è resta bloccato in attesa di ricevere la risposta, completa dei dati attesi, dall'erogatore
  - *Non Bloccante*: il fruitore non resta bloccato dopo aver inviato la richiesta, se non per ricevere una notifica di presa in carico. Per ottenere la risposta sarà necessario effettuare una distinta interazione, prevista nello scenario del servizio.
  - *Accesso CRUD*: pattern orientato alle risorse, utilizzabile solo su tecnologia REST, dove le API vengono utilizzate per eseguire operazioni di tipo CRUD - Create, Read, Update, Delete su risorse del dominio di interesse.
- *Sicurezza Canale*: gestione della sicurezza inerente il canale di comunicazione tra i domini fruitore ed erogatore. La specifica prevede i seguenti due pattern:
  - *[ID_AUTH_CHANNEL_01] Direct Trust Transport-Level Security*: comunicazione basata sul canale TLS dopo aver effettuato il trust del certificato X509 fornito dal dominio erogatore.
  - *[ID_AUTH_CHANNEL_02] Direct Trust mutual Transport-Level Security*: comunicazione basata sul canale TLS dopo aver effettuato il trust dei certificati X509, del fruitore e dell'erogatore, nella modalità di mutua autenticazione.
- *Sicurezza Messaggio*: gestione della sicurezza inerente lo scambio di informazioni tra le applicazioni agli estremi del flusso di comunicazione. I pattern di sicurezza previsti si distinguono per il caso SOAP e per quello REST:
  - *[ID_AUTH_SOAP_01 o ID_AUTH_REST_01] Direct Trust con certificato X.509 su SOAP o REST*: Tramite la validazione del certificato X509, inserito dall'applicazione mittente nel token di sicurezza della richiesta, l'applicativo destinatario verifica la corrispondenza delle identità e la validità del messaggio, prima di procedere con la produzione della risposta.
  - *[ID_AUTH_SOAP_02 o ID_AUTH_REST_02] Direct Trust con certificato X.509 su SOAP o REST con unicità del messaggio/token*: estensione del pattern precedente con l'aggiunta di un meccanismo di filtro che impedisce il processamento di un messaggio duplicato.
  - *[INTEGRITY_SOAP_01 o INTEGRITY_REST_01] Integrità del payload del messaggio SOAP o REST*: pattern che estende i precedenti aggiungendo la gestione della firma del payload come verifica di integrità del messaggio ricevuto.
- *URL di Invocazione API*: le linee guida richiedono una indicazione esplicita della tecnologia utilizzata (REST o SOAP) e la versione. Le url con cui vengono esposte le API su GovWay soddisfano entrambi i requisiti.

Tutti questi concetti sono stati recepiti e gestiti nelle maschere di configurazione della govwayConsole, adottando il profilo ModI. Le sezioni seguenti illustrano in dettaglio gli elementi di configurazione integrativi rispetto al profilo API Gateway.

3.2 Sicurezza Canale

I pattern di sicurezza a livello del canale riguardano le modalità di trasporto dei messaggi tra il dominio fruitore e quello erogatore. La specifica tenica del Modello di Interoperabilità prevede, per questa tipologia, i seguenti due pattern:

- [ID_AUTH_CHANNEL_01] Direct Trust Transport-Level Security: comunicazione basata sul canale SSL dopo aver effettuato il trust del certificato X509 fornito dal dominio erogatore.
- [ID_AUTH_CHANNEL_02] Direct Trust mutual Transport-Level Security: comunicazione basata sul canale SSL dopo aver effettuato il trust dei certificati X509, del fruitore e dell'erogatore, nella modalità di mutua autenticazione.

Il concetto di ente/dominio, previsto dalle specifiche del Modello di Interoperabilità, viene riportato su quello di Soggetto nell'ambito delle entità di configurazione di GovWay.

Vediamo nelle sezioni seguenti come si possono effettuare le configurazioni per i pattern di sicurezza canale.

3.2.1 [ID_AUTH_CHANNEL_01] Direct Trust Transport-Level Security

Questo pattern di sicurezza prevede l'utilizzo del canale HTTPS, per le comunicazioni sul confine tra i due domini, con validazione del certificato dell'ente destinatario della comunicazione.

Descriviamo di seguito i passi di configurazione da effettuare:

- La creazione della relativa API prevede che nella sezione «ModI», elemento «Sicurezza Canale», venga selezionato il pattern «ID_AUTH_CHANNEL_01» come indicato in Fig. 3.1.

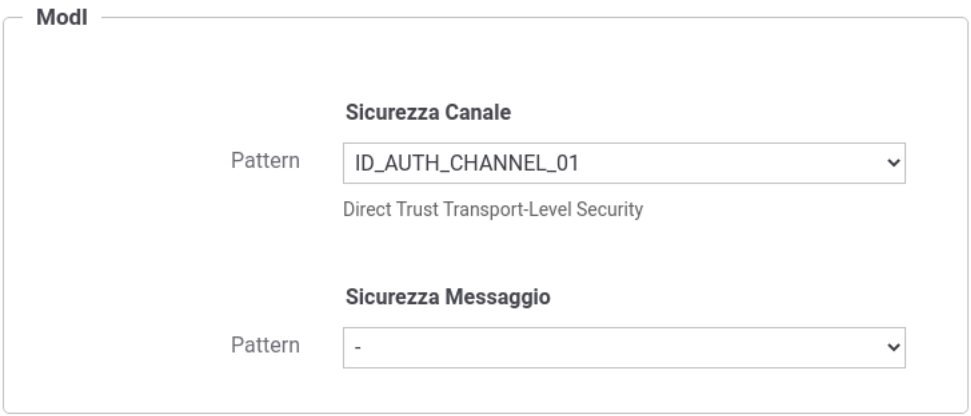


Fig. 3.1: Selezione del pattern «ID_AUTH_CHANNEL_01» per l'API

- Nel caso si voglia configurare una fruizione, le maschere di configurazione terranno conto degli aspetti di sicurezza sul canale garantendo che l'endpoint specificato nel connettore di uscita sia di tipo HTTPS, indipendentemente dal pattern adottato nella API (TLS sempre obbligatorio). L'autenticazione HTTPS può essere gestita opzionalmente da GovWay o, in alternativa, delegata alla configurazione della JVM sull'application server. Per la gestione in GovWay sono disponibili i campi per la configurazione HTTPS, lasciando opzionalmente la possibilità di impostare l'autenticazione client (vedi sez. *Autenticazione Https*).
- Nel caso si voglia configurare una erogazione, il pattern di sicurezza «ID_AUTH_CHANNEL_01» impatta sulla configurazione del Controllo Accessi, previsto nella configurazione specifica dell'erogazione:

– La sezione «Autenticazione Canale» è impostata a «HTTPS» ammettendo il flag «Opzionale» (Fig. 3.2).

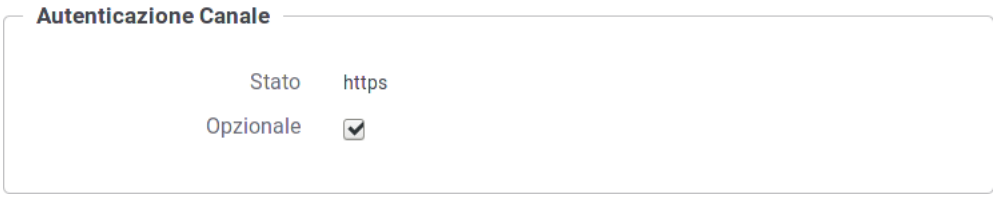


Fig. 3.2: Autenticazione Canale HTTPS con flag opzionale

– La sezione «Autorizzazione Canale» è per default disabilitata (Fig. 3.3). Abilitando tale sezione sarà possibile inserire i criteri di autorizzazione, come descritto nella sez. *Autorizzazione*, con la differenza che in questo caso le politiche saranno riferite esclusivamente ai soggetti censiti in configurazione (e non gli applicativi, per i quali si rimanda alla sez. *Sicurezza Messaggio*).

Fig. 3.3: Autorizzazione Canale Disabilitata

3.2.2 [ID_AUTH_CHANNEL_02] Direct Trust mutual Transport-Level Security

Questo pattern di sicurezza prevede l'utilizzo del canale HTTPS con autenticazione client, per le comunicazioni sul confine tra i due domini, con reciproca validazione dei certificati degli enti in gioco.

Descriviamo di seguito i passi di configurazione da effettuare:

- La creazione della relativa API prevede che nella sezione «ModI», elemento «Sicurezza Canale», venga selezionato il pattern «ID_AUTH_CHANNEL_02» come indicato in Fig. 3.4.

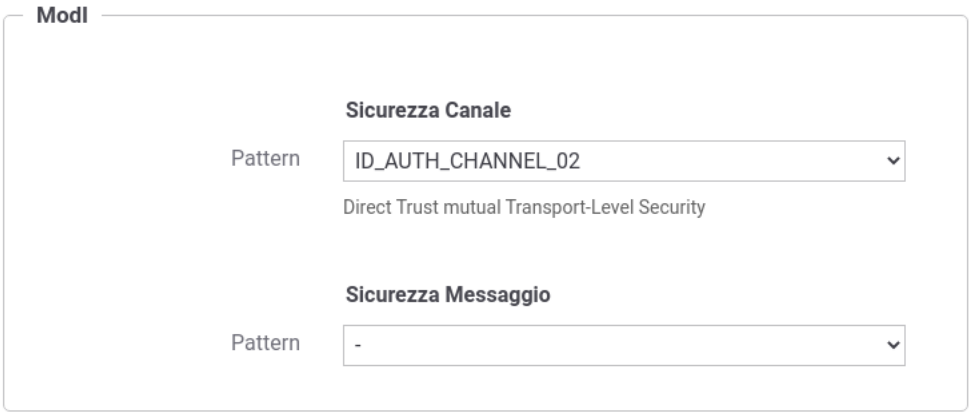


Fig. 3.4: Selezione del pattern «ID_AUTH_CHANNEL_02» per l'API

- Nel caso si voglia configurare una fruizione, le maschere di configurazione terranno conto degli aspetti di sicurezza sul canale garantendo che l'endpoint specificato nel connettore di uscita sia di tipo HTTPS, indipendentemente dal pattern adottato nella API (TLS sempre obbligatorio). L'autenticazione HTTPS può essere gestita

opzionalmente da GovWay o, in alternativa, delegata alla configurazione della JVM sull'application server. Per la gestione in GovWay sono disponibili i campi per la configurazione HTTPS, con l'obbligo di impostare l'autenticazione client (vedi sez. *Autenticazione Https*).

- Nel caso si voglia configurare una erogazione, il pattern di sicurezza «ID_AUTH_CHANNEL_02» impatta sulla configurazione del Controllo Accessi, previsto nella configurazione specifica dell'erogazione:
  - La sezione «Autenticazione Canale» è impostata forzatamente a «HTTPS» (Fig. 3.5).

Fig. 3.5: Autenticazione Canale HTTPS

  - Nella sezione «Autorizzazione Canale» è possibile attivare l'autorizzazione per richiedente inserendo gli identificativi dei soggetti autorizzati tra quelli identificati tramite il certificato SSL (Fig. 3.6). Abilitando tale sezione sarà possibile inserire i criteri di autorizzazione, come descritto nella sez. *Autorizzazione*, con la differenza che in questo caso le politiche saranno riferite esclusivamente ai soggetti censiti in configurazione (e non gli applicativi, per i quali si rimanda alla sez. *Sicurezza Messaggio*).

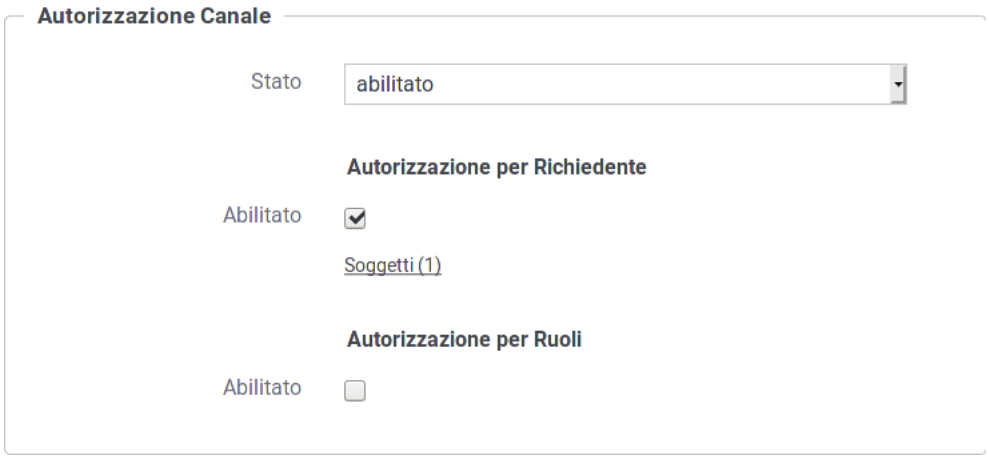


Fig. 3.6: Autorizzazione Canale su soggetti

3.3 Sicurezza Messaggio

Il pattern di sicurezza sul messaggio definisce le modalità di comunicazione dei messaggi tra componenti interne ai domini delle entità coinvolte. Tali pattern sono distinti per il caso SOAP e per quello REST:

- *[ID_AUTH_SOAP_01 o ID_AUTH_REST_01] Direct Trust con certificato X.509 su SOAP o REST*: Tramite la validazione del certificato X509, inserito dall'applicazione mittente nel token di sicurezza, l'applicativo destinatario verifica la corrispondenza delle identità e la validità del messaggio, prima di procedere con il processamento del messaggio.
- *[ID_AUTH_SOAP_02 o ID_AUTH_REST_02] Direct Trust con certificato X.509 su SOAP o REST con unicità del messaggio/token*: estensione del pattern precedente con l'aggiunta di un meccanismo di filtro che impedisce il processamento di un messaggio duplicato.

- *[INTEGRITY_SOAP_01 o INTEGRITY_REST_01] Integrità del payload del messaggio SOAP o REST*: pattern che estende i precedenti aggiungendo la gestione della firma del payload come verifica di integrità del messaggio ricevuto.

Le applicazioni di un dominio interno o esterno, descritte negli scenari del Modello di Interoperabilità, vengono rappresentate in GovWay tramite la registrazione di Applicativi come entità di configurazione. In accordo al modello di GovWay, ciascun applicativo è associato al soggetto di riferimento che, nell'ottica ModI, rappresenta il dominio di appartenenza.

Per quanto concerne le fruizioni, le richieste che provengono dagli applicativi interni del dominio e sono dirette verso altre amministrazioni vengono arricchite del token di sicurezza "ModI" associato all'operazione invocata. Gli applicativi vengono identificati attraverso una delle modalità di autenticazione previste da GovWay (vedi sez. *Autenticazione Trasporto*) ed una volta identificato viene utilizzato il certificato X509 associatogli in fase di registrazione da utilizzare per effettuare la firma del token di sicurezza "ModI" (Fig. 3.7).

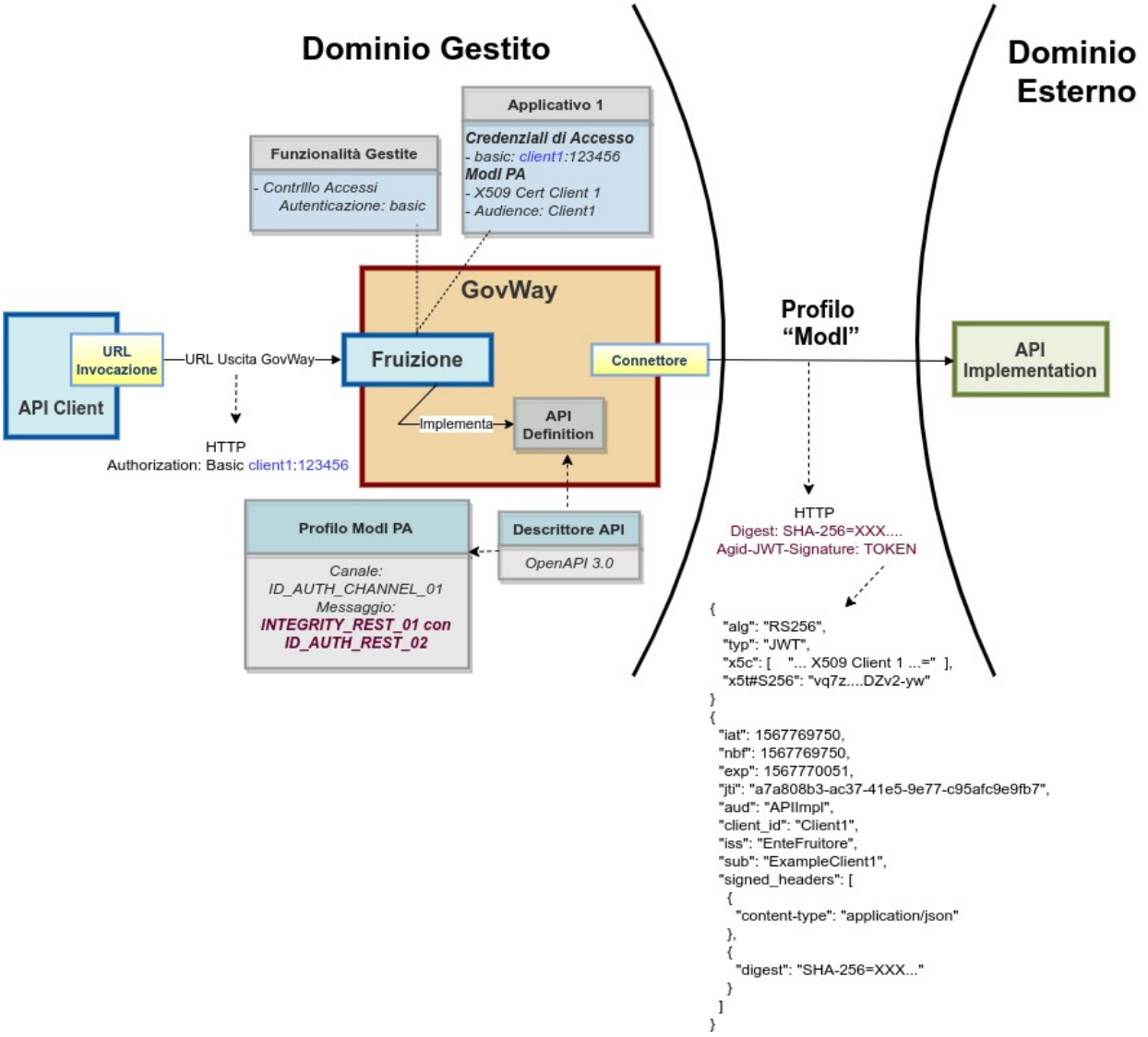


Fig. 3.7: Fruizione con Profilo di Interoperabilità "ModI"

Nelle erogazioni invece, le richieste provengono da amministrazioni esterne al dominio e sono dirette ad applicativi interni. Prima di procedere con l'inoltro della richiesta verso il backend interno, GovWay valida il token di sicurezza ricevuto rispetto al pattern associato all'operazione invocata: verifica firma, validazione temporale, filtro duplicati, verifica integrità del messaggio . . . (Fig. 3.8)

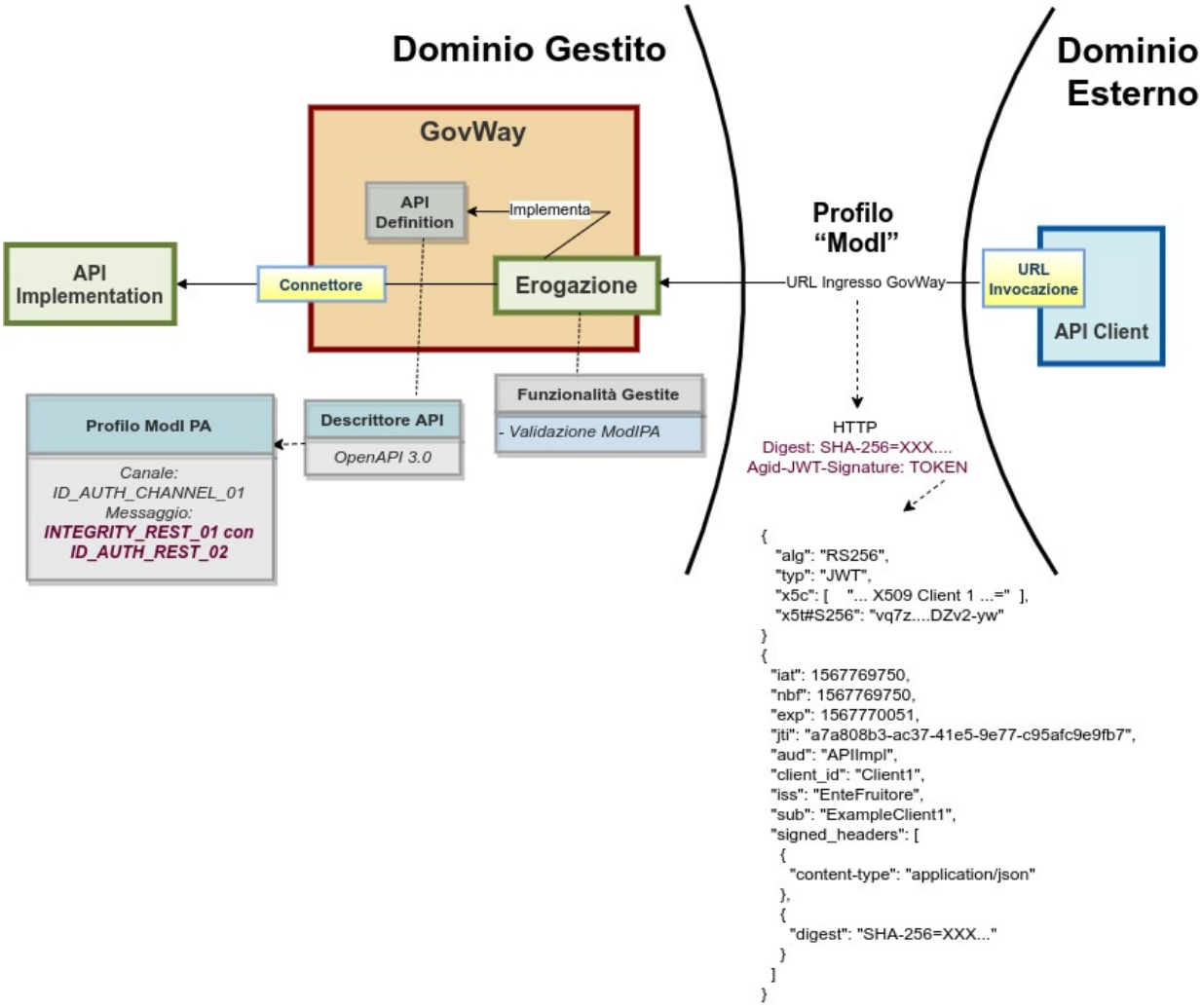


Fig. 3.8: Erogazione con Profilo di Interoperabilità "ModI"

Vediamo nelle sezioni seguenti come si possono effettuare le configurazioni relative ai pattern di sicurezza messaggio.

3.3.1 Passi preliminari di configurazione

In questa sezione viene indicato come effettuare una configurazione iniziale dei seguenti aspetti di gestione dei certificati X509 utilizzati all'interno dei token di sicurezza "ModI".

TrustStore per la validazione dei Certificati

Per le richieste, provenienti da amministrazioni esterne, GovWay deve validare il certificato presente all'interno del token di sicurezza al fine di verificare che sia rilasciato da una della CA conosciute, che non sia scaduto e che non sia stato eventualmente revocato. Per poter effettuare la validazione, deve essere configurato opportunamente GovWay per quanto riguarda le seguenti proprietà presenti nel file "/etc/govway/modipa_local.properties" (dove si assume che "/etc/govway" sia la directory di configurazione indicata in fase di installazione) tutte con prefisso "org.openspcoop2.protocol.modipa.sicurezzaMessaggio.certificati.":

- *trustStore.path* (obbligatorio): indica il path su file system di un trustStore contenente le CA conosciute.
- *trustStore.tipo* (obbligatorio): indica il tipo di trustStore (JKS)
- *trustStore.password* (obbligatorio): password per accedere al trustStore
- *trustStore.crls* (opzionale): permette di indicare un elenco, separato da virgola, di file CRL.

La configurazione sopra indicata rappresenta la configurazione di default che verrà proposta durante la gestione di una erogazione o di una fruizione. È sempre possibile ridefinire per ogni singola API tale configurazione

---

Nota: TrustStore delle comunicazioni HTTPS

Nei pattern di sicurezza per API REST, dove il riferimento al certificato utilizzato viaggia tramite il claim "x5u", è possibile che GovWay debba effettuare il download del certificato tramite url https che espongono certificati server non validabili tramite le CA note. In tale contesto è possibile configurare un trustStore personalizzato agendo sulle proprietà presenti nel file "/etc/govway/modipa_local.properties" in maniera simile al trustStore dei certificati. Tali proprietà possiedono il prefisso "org.openspcoop2.protocol.modipa.sicurezzaMessaggio.ssl.".

KeyStore per la firma della Risposte

Nella figura Fig. 3.7 della sezione *Sicurezza Messaggio* è stato descritto come GovWay utilizzerà la chiave privata associata all'applicativo interno che ha scaturito la richiesta per firmare il token di sicurezza aggiunto al messaggio in uscita dal dominio di gestione. Per quanto concerne invece le risposte che GovWay processa in una erogazione, la chiave privata utilizzata per firmare il token di sicurezza aggiunto alla risposta viene preso da una configurazione di default descritta di seguito. È sempre possibile ridefinire per ogni singola API tale configurazione.

Per poter firmare i token di sicurezza delle risposte, deve essere configurato opportunamente GovWay per quanto riguarda le seguenti proprietà presenti nel file "/etc/govway/modipa_local.properties" tutte con prefisso "org.openspcoop2.protocol.modipa.sicurezzaMessaggio.certificati.":

- *keyStore.path* (obbligatorio): indica il path su file system di un keyStore contenente la chiave privata.
- *keyStore.tipo* (obbligatorio): indica il tipo di trustStore (JKS)
- *keyStore.password* (obbligatorio): password per accedere al keyStore
- *key.alias* (obbligatorio): alias della chiave privata all'interno del keyStore.
- *key.password* (obbligatorio): password della chiave privata all'interno del keyStore.

3.3.2 [ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509

Nota: La sigla che identifica il pattern di sicurezza messaggio varia a seconda se l'API sia di tipo REST, per cui la sigla corrisponde a *ID_AUTH_REST_01*, o SOAP dove viene utilizzata la sigla *ID_AUTH_SOAP_01*.

L'adozione di questo pattern consente, alla ricezione di un messaggio, di validare il certificato fornito dall'applicativo mittente, la porzione di messaggio firmata, la validità temporale nonché la corrispondenza del destinatario della comunicazione.

Nel processo di configurazione, per i servizi con questo pattern, la registrazione delle API prevede che nella sezione «ModI», elemento «Sicurezza Messaggio», venga selezionato il pattern «ID_AUTH_REST_01» per API REST o «ID_AUTH_SOAP_01» per API SOAP come indicato in Fig. 3.9 e Fig. 3.10.

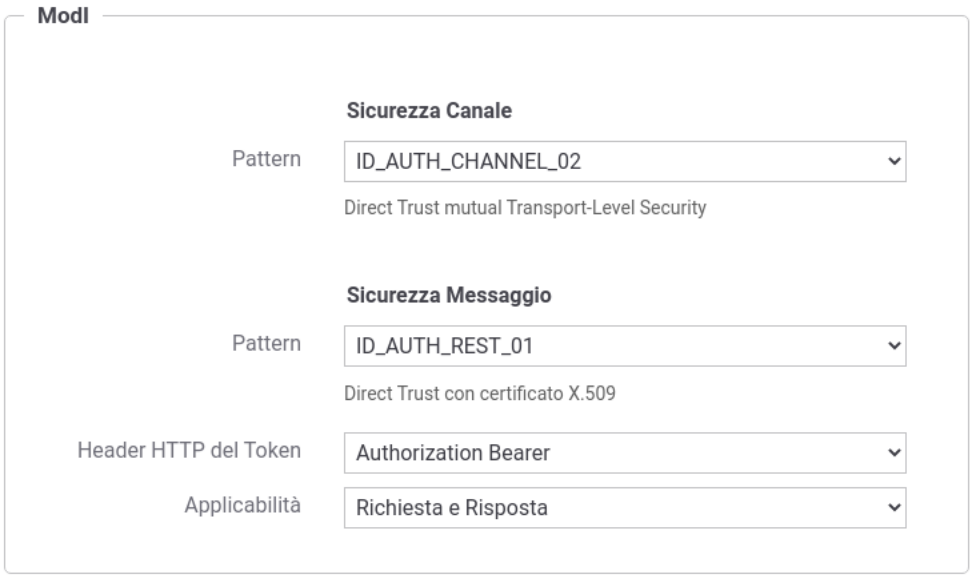


Fig. 3.9: Pattern di sicurezza messaggio «ID_AUTH_REST_01» per l'API

Le voci "Header HTTP del Token" (presente solamente su API di tipo REST) e "Applicabilità" consentono di personalizzare l'header HTTP utilizzato e di indicare se il pattern di sicurezza verrà attuato sia sulla richiesta che sulla risposta. Maggiori informazioni vengono fornite nella sezione "*Funzionalità Avanzate*".

Nel contesto della configurazione della specifica operation/risorsa è presente anche la sezione «Sicurezza Messaggio» che consente di intervenire sul pattern di sicurezza messaggio in modo puntuale. È quindi possibile lasciare l'impostazione del pattern al valore già stabilito a livello della API, oppure decidere di ridefinirlo andando a fornire una configurazione specifica per la singola operation/risorsa come indicato in Fig. 3.9.

Il processo prosegue con alcune differenze in base al tipo di servizio che si vuole configurare.

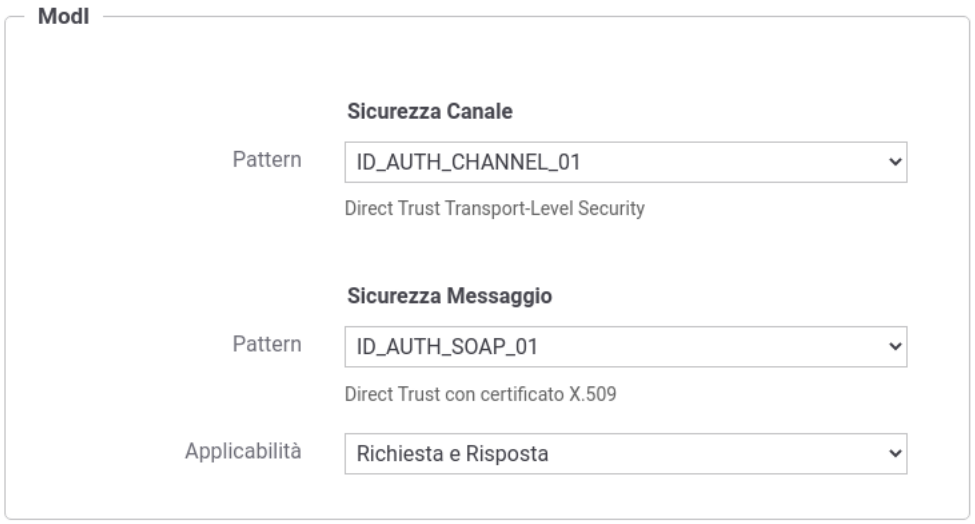


Fig. 3.10: Pattern di sicurezza messaggio «ID_AUTH_SOAP_01» per l'API

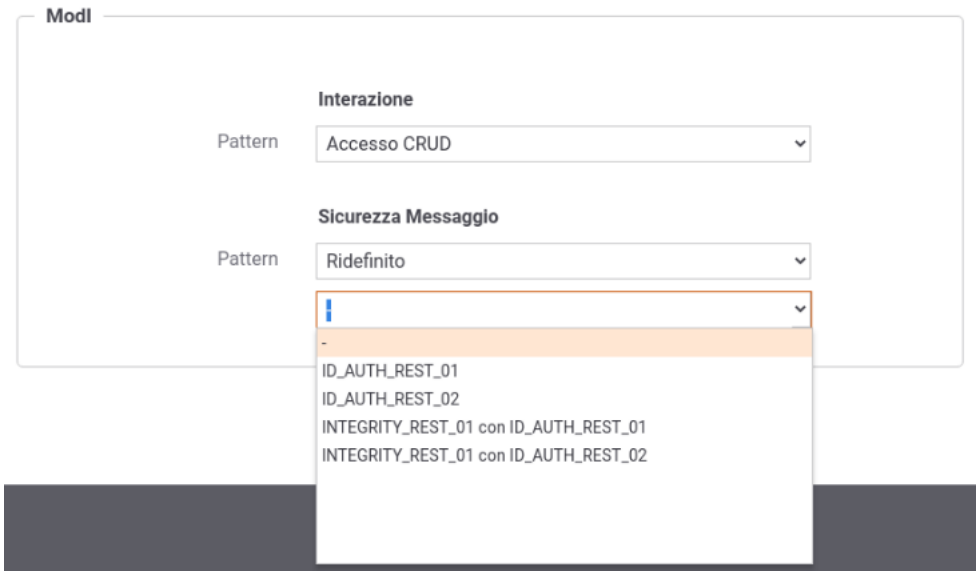


Fig. 3.11: Pattern di sicurezza messaggio ridefinito per una risorsa dell'API

Fruizione

Le richieste che provengono dagli applicativi interni del dominio e sono dirette verso altre amministrazioni verranno arricchite del token di sicurezza "ModI" previsto dall'operazione invocata, come indicato precedentemente nella sezione *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*.

Per la configurazione delle fruizioni con i pattern di sicurezza messaggio è necessario registrare ciascun applicativo interno coinvolto al fine principale di associargli una chiave privata e un certificato X509 che GovWay utilizza per firmare il token di sicurezza "ModI" prodotto. Gli applicativi vengono identificati da GovWay tramite una delle modalità di autenticazione supportate descritte nella sezione *Autenticazione Trasporto* (Fig. 3.12).

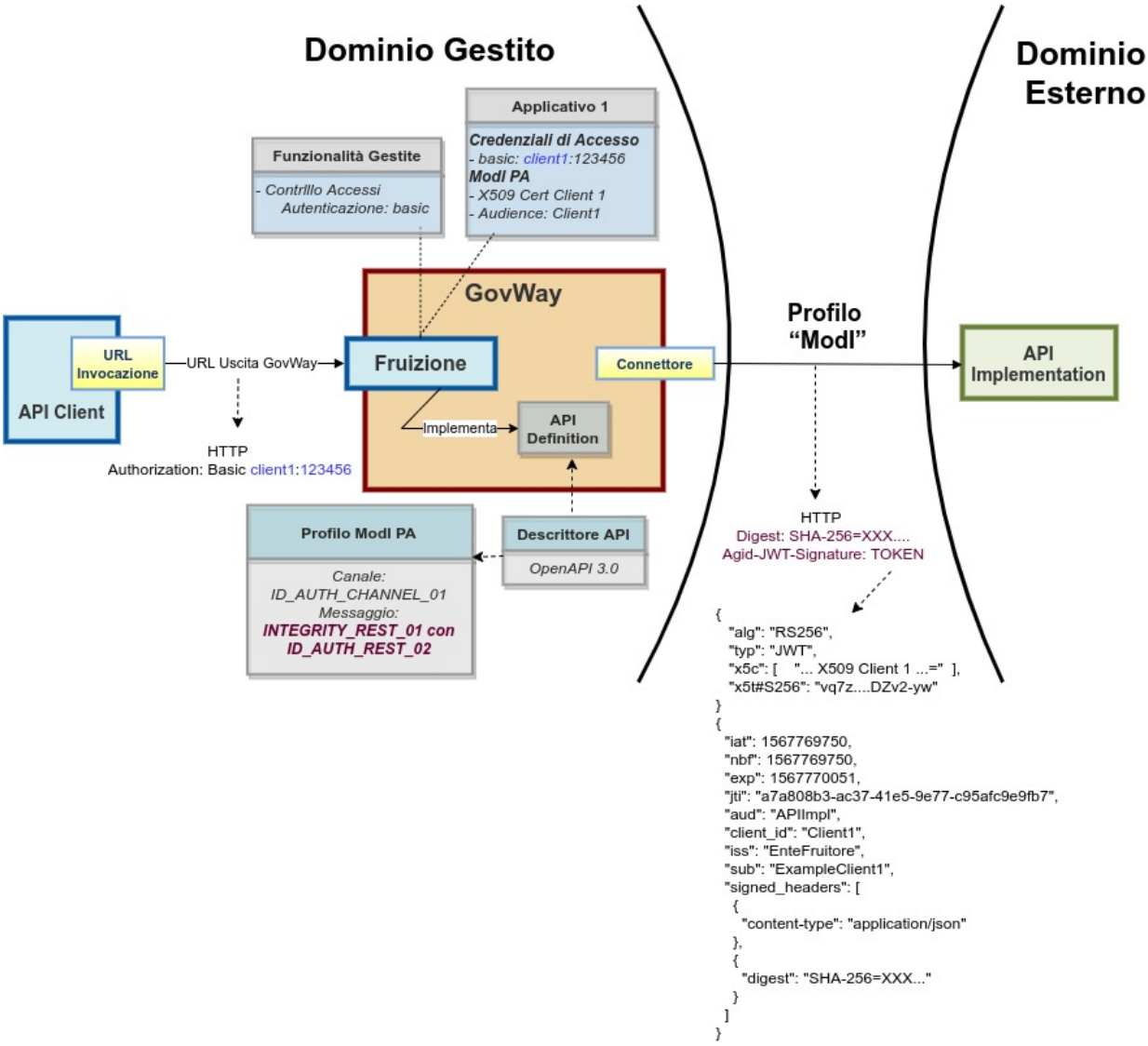


Fig. 3.12: Fruizione con Profilo di Interoperabilità "ModI"

La registrazione dell'applicativo avviene come già descritto nella sez. *Creazione di un applicativo*. In questo contesto sarà necessario specificare il dominio «Interno» dell'applicativo e procedere all'inserimento dei dati nel form «ModI» (Fig. 3.13).

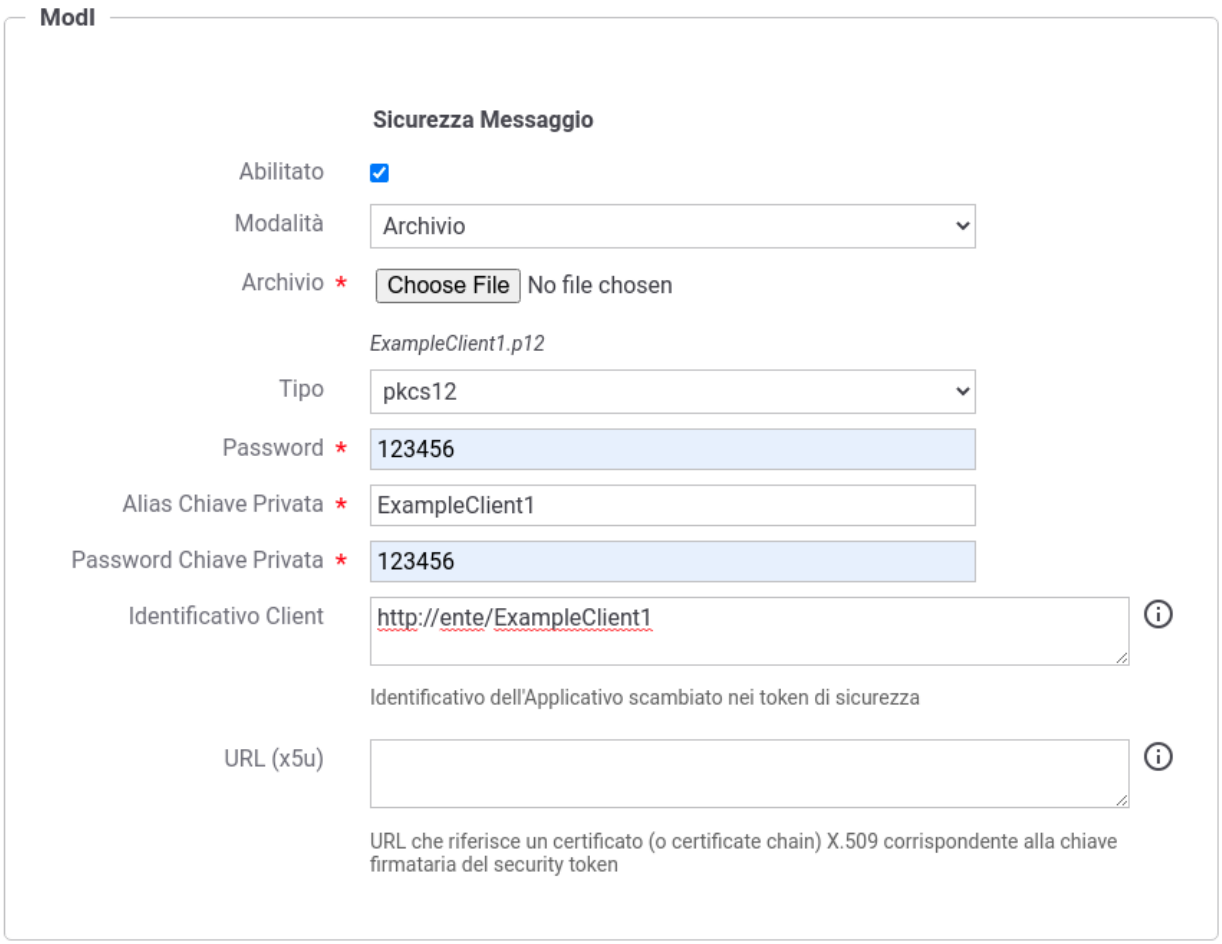


Fig. 3.13: Dati ModI relativi ad un applicativo interno

I dati da inserire sono:

- *Archivio*: il file che corrisponde al keystore contenente la chiave privata utilizzata per la firma dei messaggi
- *Tipo*: il formato del keystore (jks, pkcs12)
- *Password*: la password per l'accesso al keystore
- *Alias Chiave Privata*: l'alias con cui è riferita la chiave privata nel keystore
- *Password Chiave Privata*: la password della chiave privata
- *Identificativo Client*: identificativo dell'applicativo utilizzato per valorizzare nel token di sicurezza di una richiesta il claim "client_id" per API REST e l'header "wsa:From" per API SOAP (**Attenzione**: se non definito viene utilizzato il nome dell'applicativo). Se è abilitata la funzionalità "Verifica Audience / WSAddressing To" nella configurazione della sicurezza della risposta verrà inoltre verificato che nel token di sicurezza della risposta ricevuto vi sia un claim "aud" per API REST o un header "wsa:To" per API SOAP che possiede un valore identico all'identificato fornito.
- *URL (x5u)*: URL che riferisce un certificato (o certificate chain) X.509 corrispondente alla chiave firmataria del security token. Deve essere obbligatoriamente definito se l'applicativo fruisce di API REST configurate per generare un token di sicurezza tramite il claim "x5u"

L'interfaccia per la creazione della fruizione, basata su una API con pattern «ID_AUTH_REST_01» (o «ID_AUTH_SOAP_01»), presenta le sezioni «ModI - Richiesta» e «ModI - Risposta»:

- ModI - Richiesta (Fig. 3.14): la maschera relativa alla richiesta prevede la configurazione del meccanismo di firma digitale del messaggio, ad opera dell'applicativo mittente, e la produzione del relativo token di sicurezza:
  - Algoritmo: l'algoritmo che si vuole utilizzare per la firma digitale del messaggio
  - Riferimento X.509: il metodo da utilizzare per l'inserimento del certificato dell'applicativo nel token di sicurezza. I valori possibili sono (differenziati per il caso REST e SOAP) quelli previsti nelle Linee Guida di Interoperabilità:
  - Certificate Chain: se è stata selezionata la modalità "x5c", è possibile indicare se nel token di sicurezza verrà incluso solo il certificato utilizzato per la firma o l'intera catena.
  - Time to Live: tempo di validità del token prodotto (in secondi)
  - Audience: identificativo dell'applicativo destinatario da indicare come audience nel token di sicurezza; se non viene indicato alcun valore verrà utilizzato la url del connettore. Il valore fornito può contenere parti dinamiche risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*).
- ModI - Risposta (Fig. 3.15): la maschera relativa alla risposta prevede la configurazione del meccanismo di validazione del token ricevuto da parte dell'applicativo destinatario:
  - Riferimento X.509: il metodo per la localizzazione del certificato del destinatario nel messaggio di risposta. Si può mantenere la medesima impostazione prevista per il messaggio di richiesta o ridefinirla.
  - TrustStore Certificati: Riferimento al truststore che contiene le CA, i certificati e le CRL da utilizzare per poter verificare i token di sicurezza ricevuti nelle risposte. È possibile mantenere l'impostazione di default che è stata fornita al momento dell'installazione del prodotto, oppure definire un diverso riferimento (opzione «Ridefinito») fornendo Path, Tipo, Password del TrustStore e CRL.
  - Verifica Audience: Se l'opzione è abilitata, viene effettuata la verifica che il campo Audience, presente nel token di sicurezza della risposta, corrisponda al valore presente nel campo successivo, se indicato, o altrimenti a quello configurato nell'applicativo mittente nella voce "Identificativo Client".

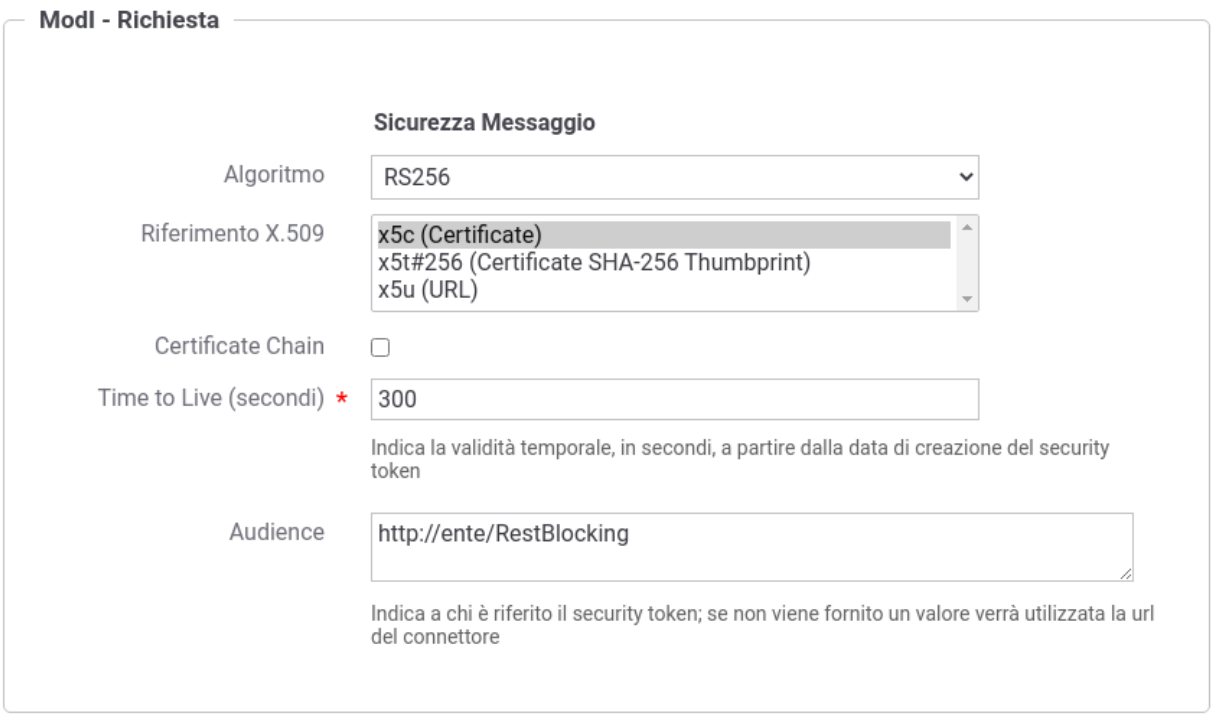


Fig. 3.14: Dati per la configurazione della sicurezza messaggio sulla richiesta di una fruizione

Erogazione

Nelle erogazioni, le richieste provengono da amministrazioni esterne al dominio e sono dirette ad applicativi interni. Prima di procedere con l'inoltro della richiesta verso il backend interno, GovWay valida il token di sicurezza ricevuto rispetto al pattern associato all'operazione invocata (come descritto nella sezione *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*): verifica firma, validazione temporale, filtro duplicati, verifica integrità del messaggio . . . (Fig. 3.16)

Per la configurazione di erogazioni basate su una API con pattern «ID_AUTH_REST_01» (o «ID_AUTH_SOAP_01»), nella relativa maschera della govwayConsole saranno presenti le sezioni «ModI - Richiesta» e «ModI - Risposta»:

- ModI - Richiesta (Fig. 3.17): la maschera relativa alla richiesta prevede la configurazione del meccanismo di validazione del token ricevuto sul messaggio di richiesta:
  - Riferimento X.509: il metodo per la localizzazione del certificato dell'applicativo mittente nel messaggio di richiesta. Il valore fornito deve corrispondere alla scelta operata dai mittenti. I valori possibili (differenziati per il caso REST e SOAP) sono quelli previsti nella specifica AGID.
  - TrustStore Certificati: Riferimento al truststore che contiene le CA, i certificati e le CRL da utilizzare per poter verificare i token di sicurezza ricevuti nelle richieste. È possibile mantenere l'impostazione di default che è stata fornita al momento dell'installazione del prodotto, oppure definire un diverso riferimento (opzione «Ridefinito») fornendo Path, Tipo, Password del TrustStore e CRL.
  - Audience: valore del campo Audience atteso nel token di sicurezza della richiesta.
- ModI - Risposta (Fig. 3.18): la maschera prevede la configurazione del meccanismo di firma digitale del messaggio di risposta, e la produzione del relativo token di sicurezza, da inviare all'applicativo mittente:

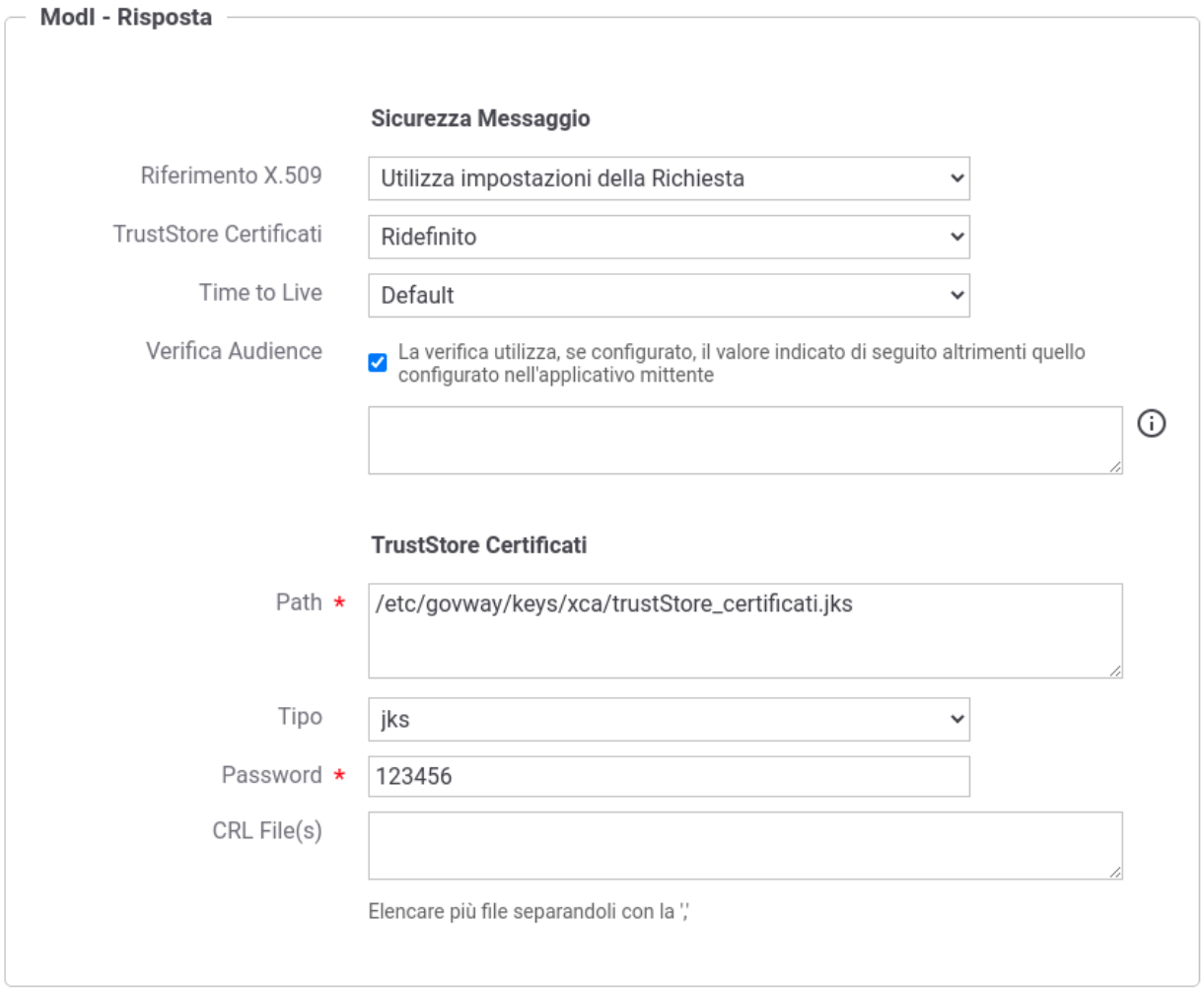


Fig. 3.15: Dati per la configurazione della sicurezza messaggio sulla risposta di una fruizione

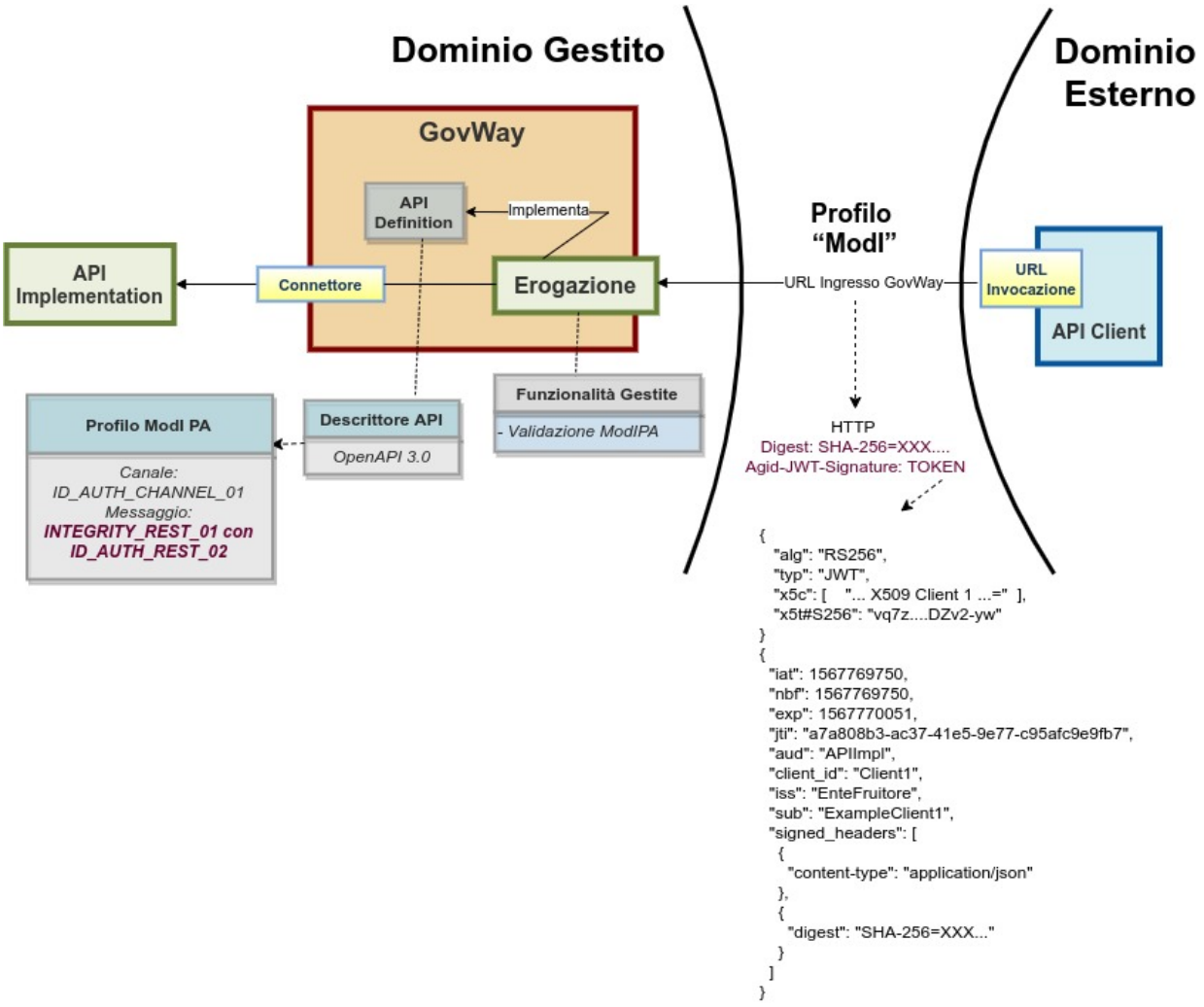


Fig. 3.16: Erogazione con Profilo di Interoperabilità "ModI"

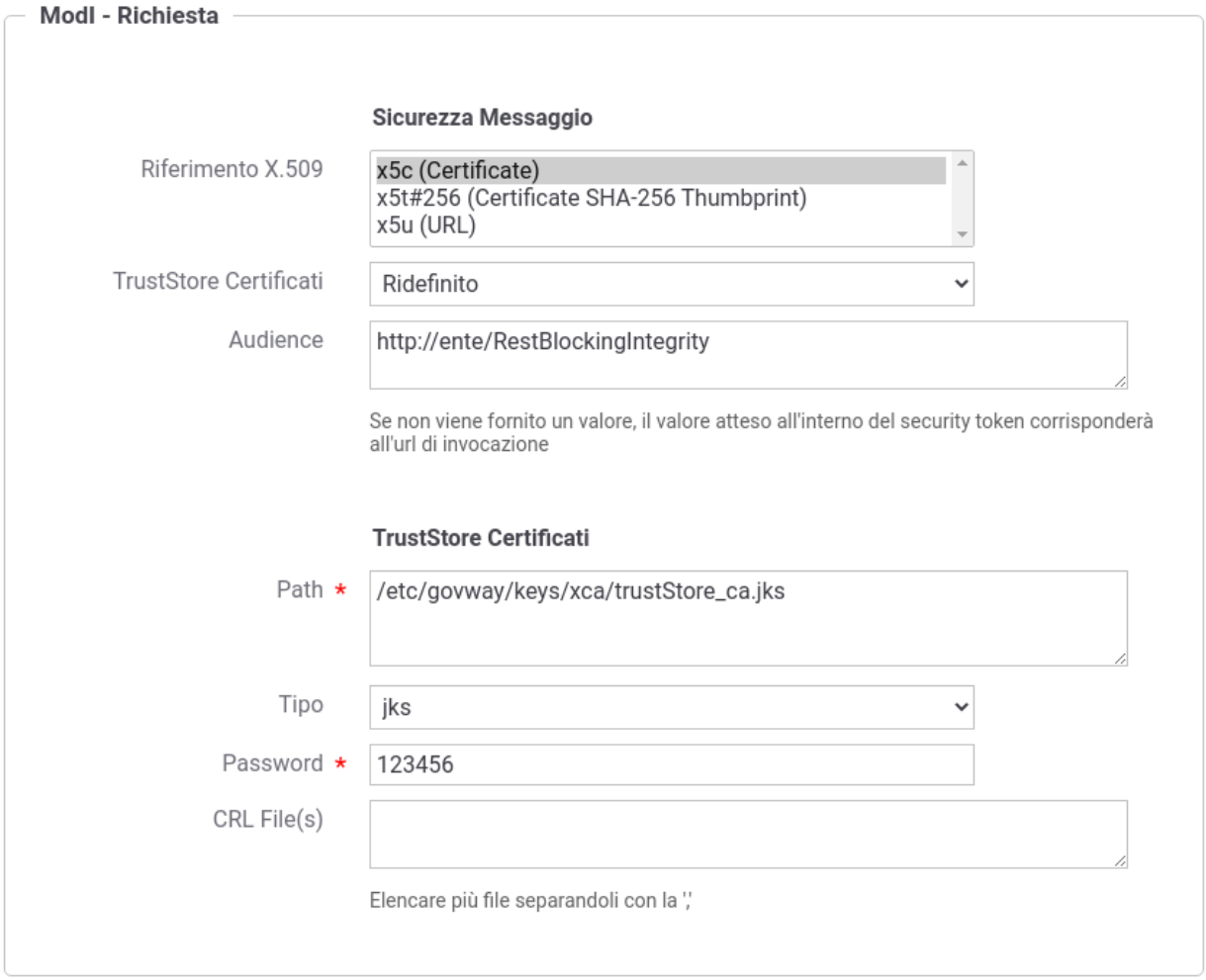


Fig. 3.17: Dati per la configurazione della sicurezza messaggio sulla richiesta di una erogazione

- Algoritmo: l'algoritmo che si vuole utilizzare per la firma digitale del messaggio di risposta
- Riferimento X.509: il metodo da utilizzare per l'inserimento del certificato nel messaggio di risposta. Si può mantenere la medesima impostazione prevista per il messaggio di richiesta o ridefinirla.
- Certificate Chain: se è stata selezionata la modalità "x5c", è possibile indicare se nel token di sicurezza verrà incluso solo il certificato utilizzato per la firma o l'intera catena.
- Keystore: il keystore da utilizzare per la firma del messaggio di risposta. È possibile mantenere il riferimento al keystore di default, fornito in fase di installazione del prodotto, oppure indicare un diverso riferimento (opzione «Ridefinito») fornendo il path sul filesystem, o in alternativa direttamente l'archivio, unitamente a Tipo, Password, Alias Chiave Privata e Password Chiave Privata.
- Time to Live (secondi): validità temporale del token prodotto.

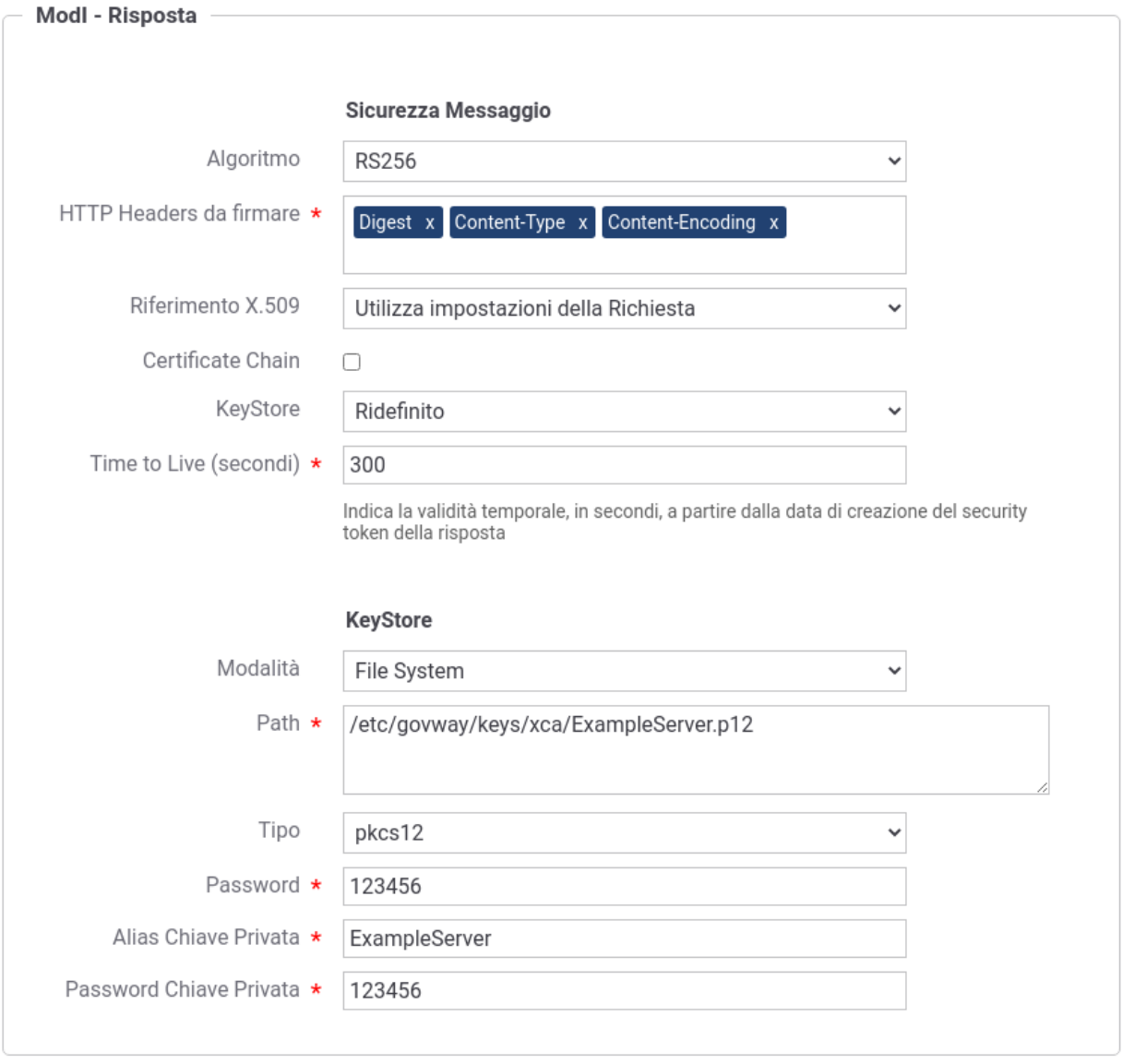


Fig. 3.18: Dati per la configurazione della sicurezza messaggio sulla risposta di una erogazione

Nel contesto dei pattern di sicurezza messaggio è possibile registrare anche gli applicativi dei domini esterni al fine di:

1. identificare puntualmente le componenti esterne coinvolte nella comunicazione abilitando le funzionalità di tracciamento e statistica per tali elementi.
2. abilitare le funzionalità di autorizzazione sugli applicativi identificando puntualmente chi autorizzare dopo il superamento del processo di autenticazione/autorizzazione canale e validazione del token di sicurezza (Fig. 3.19).

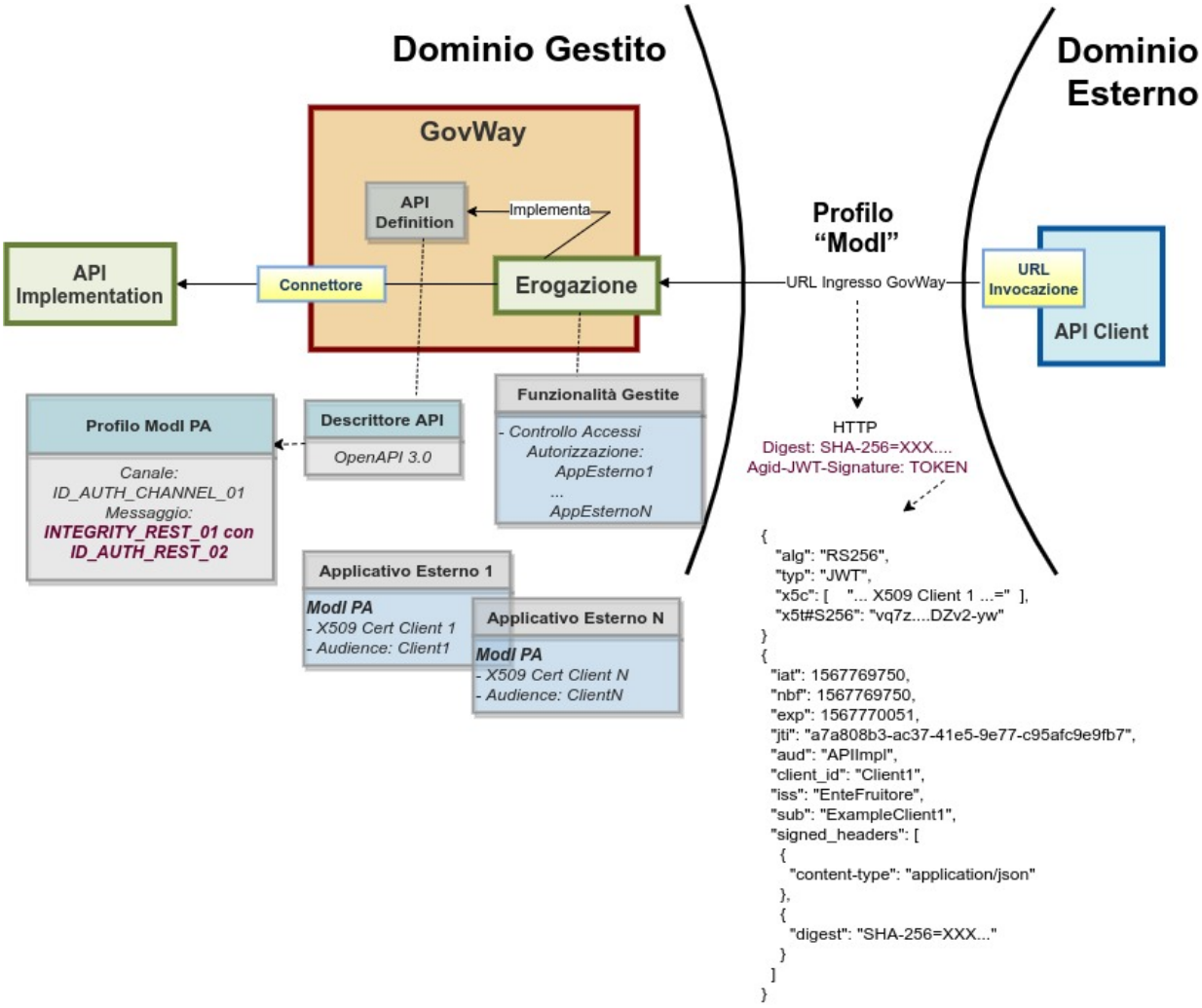


Fig. 3.19: Erogazione con Profilo di Interoperabilità "ModI" e criteri di autorizzazione puntuali

Per abilitare quanto al punto 1 è sufficiente la sola registrazione degli applicativi esterni coinvolti (Fig. 3.20).

Dopo aver indicato il dominio «Esterno» per l'applicativo, sarà necessario selezionare il soggetto che identifica il dominio esterno di riferimento.

La registrazione dell'applicativo esterno comprende anche la sezione con i dati relativi alla sicurezza messaggio (Fig. 3.21).

I dati da fornire sono:

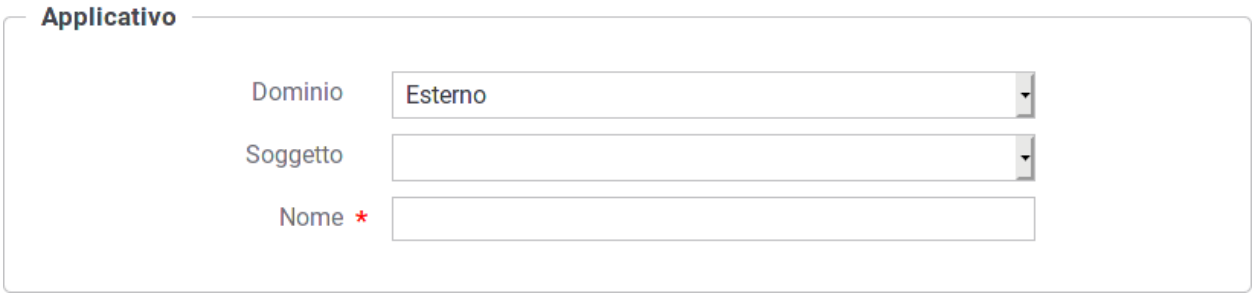


Fig. 3.20: Registrazione di un applicativo esterno

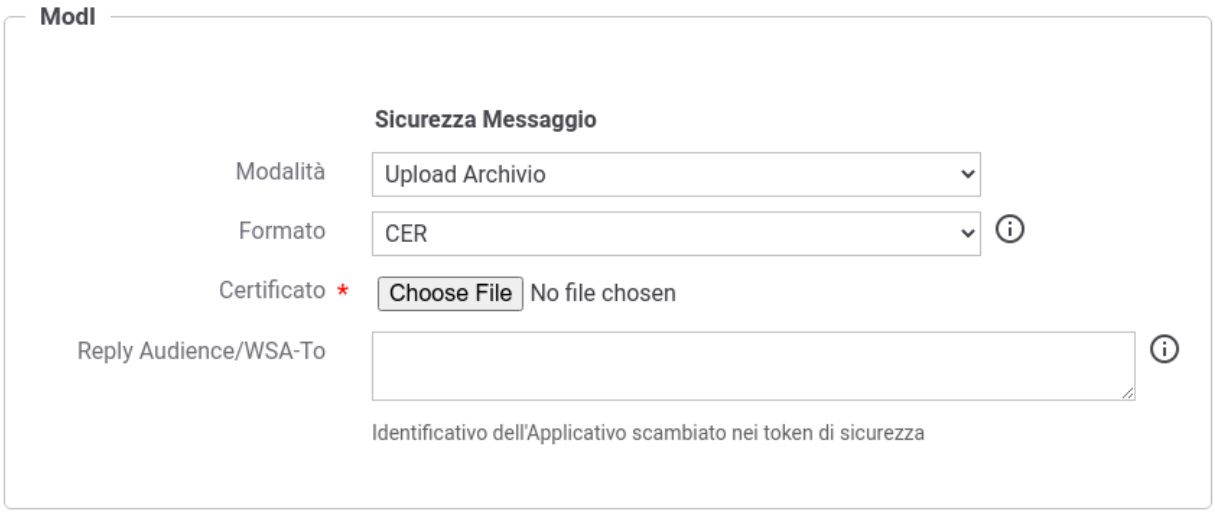


Fig. 3.21: Dati ModI relativi ad un applicativo esterno con upload del certificato

- *Modalità*: si seleziona tra il caricamento del certificato e la configurazione manuale
- Caso *Upload Archivio*:
  - *Formato*: formato dell'archivio fornito (CER, JKS; PKCS12)
  - *Certificato*: elemento per l'upload dell'archivio che contiene il certificato
  - *Reply Audience/WSA-To*: identificativo dell'applicativo da confrontare con il valore «Audience» eventualmente presente nelle richieste.
- Caso *Configurazione Manuale* (Fig. 3.22):
  - *Self Signed*: opzione per indicare se il cerfificato è self-signed oppure rilasciato da una CA
  - *Subject*: il subject del certificato
  - *Issuer*: l'issuer del certificato, nel caso in cui non sia self-signed
  - *Reply Audience/WSA-To*: identificativo dell'applicativo da confrontare con il valore «Audience» eventualmente presente nelle richieste.

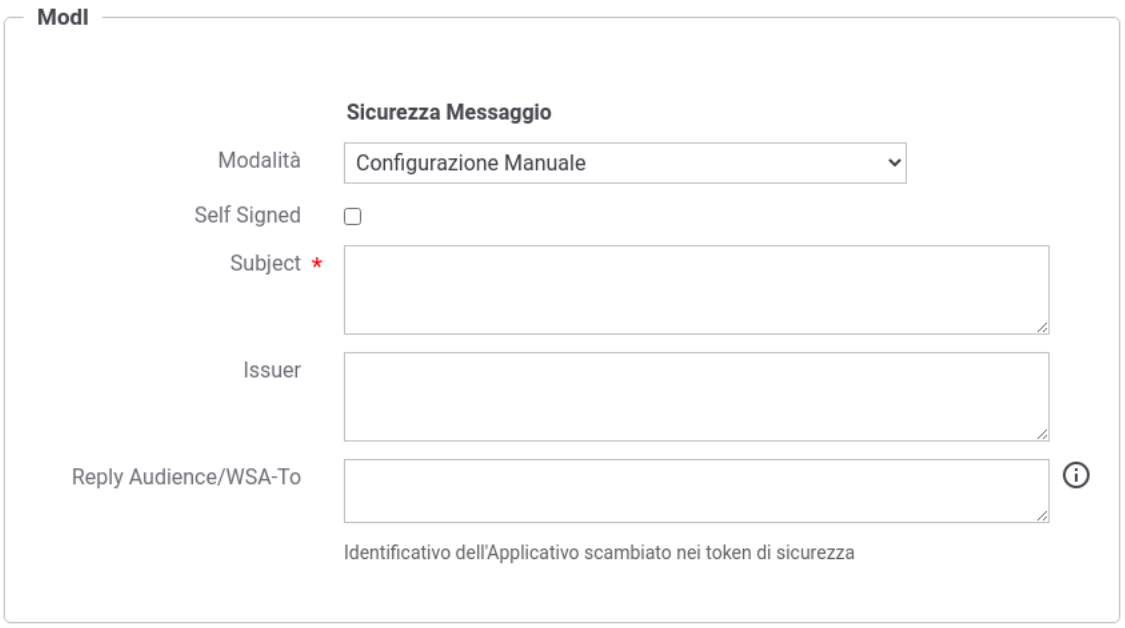


Fig. 3.22: Dati ModI relativi ad un applicativo esterno con configurazione manuale dei dati di sicurezza

Per abilitare le funzionalità di autorizzazione dei singoli applicativi (punto 2 del precedente elenco) si deve procedere alla configurazione della sezione «Controllo Accessi» relativa all'erogazione. Quando attiva la sicurezza messaggio, questa sezione conterrà il form «Autorizzazione ModI» (Fig. 3.23). Qui è possibile specificare un elenco di applicativi (esterni) autorizzati, ad accedere all'erogazione, tra quelli identificati nella fase di verifica del relativo certificato. Gli applicativi esterni saranno selezionabili tra quelli censiti nella sezione «Applicativi» (Fig. 3.23).

Nota: L'autorizzazione basata sugli identificativi degli applicativi mittenti del dominio fruitore esterno, è possibile soltanto se è stata effettuata la registrazione degli applicativi interessati, in associazione al soggetto esterno di riferimento.

Fig. 3.23: Autorizzazione di singoli applicativi per l'accesso all'erogazione

3.3.3 [ID_AUTH_SOAP_02 / ID_AUTH_REST_02] Direct Trust con certificato X.509 con unicità del messaggio/token

Nota: La sigla che identifica il pattern di sicurezza messaggio varia a seconda se l'API sia di tipo REST, per cui la sigla corrisponde a *ID_AUTH_REST_02*, o SOAP dove viene utilizzata la sigla *ID_AUTH_SOAP_02*.

Questo pattern di sicurezza presenta le medesime caratteristiche di *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*, con l'unica differenza di prevedere un meccanismo di filtro che impedisce la ricezione di messaggi duplicati da parte di ciascun ricevente.

L'attivazione di questo pattern avviene a livello della relativa API, nella sezione «ModI», elemento «Sicurezza Messaggio», selezionando il pattern «ID_AUTH_REST_02» per API REST o «ID_AUTH_SOAP_02» per API SOAP come indicato in Fig. 3.24 e Fig. 3.25.

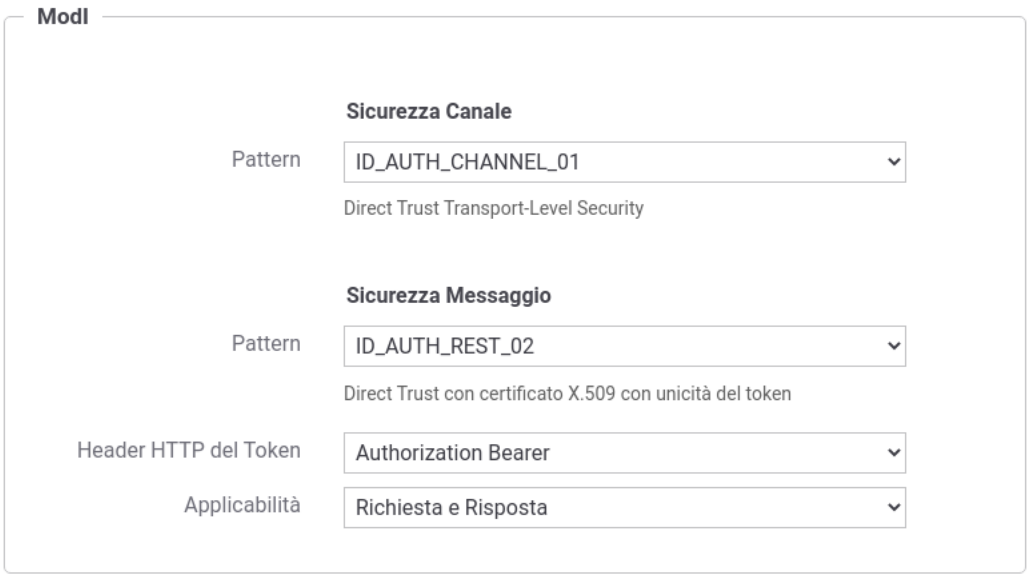


Fig. 3.24: Pattern di sicurezza messaggio «ID_AUTH_REST_02» per l'API

Le voci "Header HTTP del Token" (presente solamente su API di tipo REST) e "Applicabilità" consentono di personalizzare l'header HTTP utilizzato e di indicare se il pattern di sicurezza verrà attuato sia sulla richiesta che sulla risposta. Maggiori informazioni vengono fornite nella sezione "*Funzionalità Avanzate*".

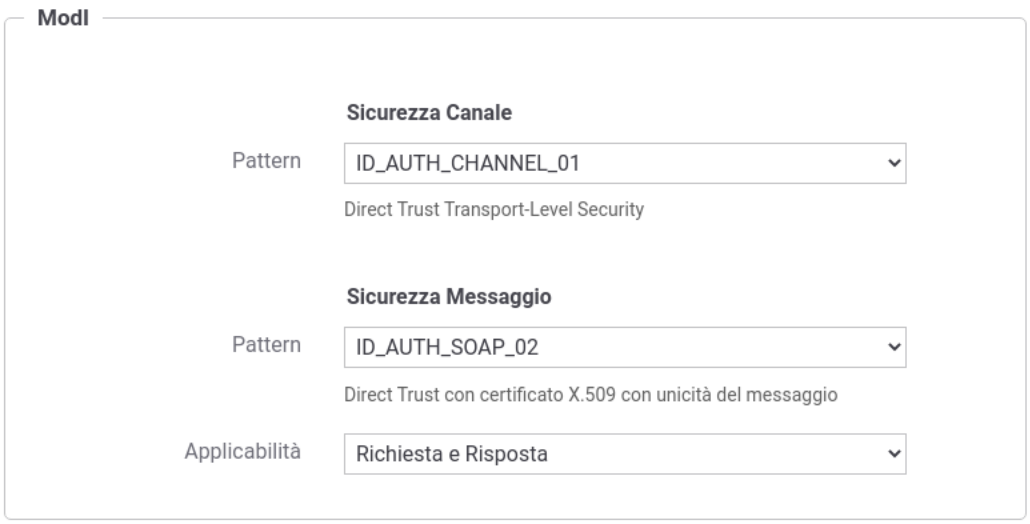


Fig. 3.25: Pattern di sicurezza messaggio «ID_AUTH_SOAP_02» per l'API

Per le configurazioni successive procedere come già descritto in precedenza per il pattern *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*.

3.3.4 [INTEGRITY_SOAP_01 / INTEGRITY_REST_01] Integrità payload del messaggio

Nota: La sigla che identifica il pattern di sicurezza messaggio varia a seconda se l'API sia di tipo REST, per cui la sigla corrisponde a *INTEGRITY_REST_01*, o SOAP dove viene utilizzata la sigla *INTEGRITY_SOAP_01*.

Questo pattern di sicurezza consente di estendere «ID_AUTH_REST_01» o «ID_AUTH_REST_02» aggiungendo un meccanismo che garantisce l'integrità del messaggio scambiato grazie all'invio, nel token di sicurezza, della firma digitale del payload.

L'attivazione di questo pattern avviene a livello della relativa API, nella sezione «ModI», elemento «Sicurezza Messaggio», selezionando il pattern «INTEGRITY_REST_01 con ID_AUTH_REST_01» nel caso si voglia estendere «ID_AUTH_REST_01», oppure il pattern «INTEGRITY_REST_01 con ID_AUTH_REST_02» nel caso si voglia estendere «ID_AUTH_REST_02» con il meccanismo di garanzia dell'integrità del payload (Fig. 3.26).

Le voci "Header HTTP del Token" (presente solamente su API di tipo REST) e "Applicabilità" consentono di personalizzare gli header HTTP utilizzati (nomi e contemporaneità) e di indicare se il pattern di sicurezza verrà attuato sia sulla richiesta che sulla risposta. Su API di tipo SOAP è possibile selezionare una "Applicatibilità" che firmi oltre al body anche gli attachments, se presenti. Maggiori informazioni vengono fornite nella sezione "*Funzionalità Avanzate*".

La voce "Informazioni Utente" consente di abilitare la funzionalità, descritta nella sezione "*Informazioni Utente*", che consente di aggiungere all'interno del token di sicurezza le informazioni sull'utente che ha effettuato la richiesta.

La voce "Digest Richiesta" consente di abilitare la funzionalità, descritta nella sezione "*Digest della Richiesta - Non ripudiabilità della trasmissione*", che consente di implementare la soluzione per la non ripudiabilità della trasmissione.

Per le configurazioni successive procedere come già descritto in precedenza per il pattern *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*.

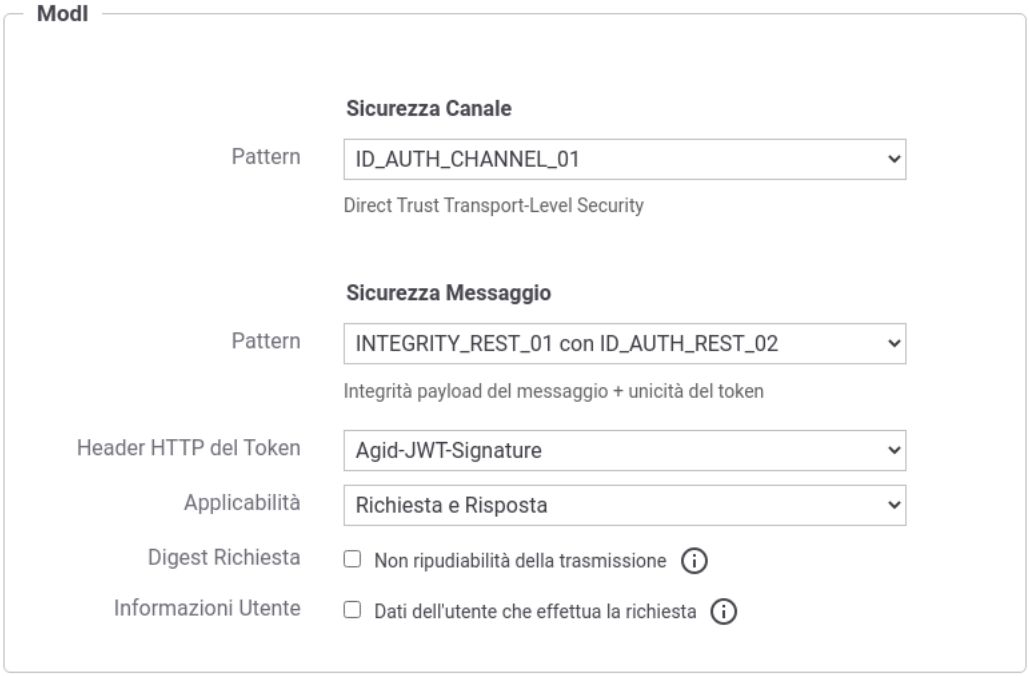


Fig. 3.26: Pattern di sicurezza messaggio «INTEGRITY_REST_01» per l'API

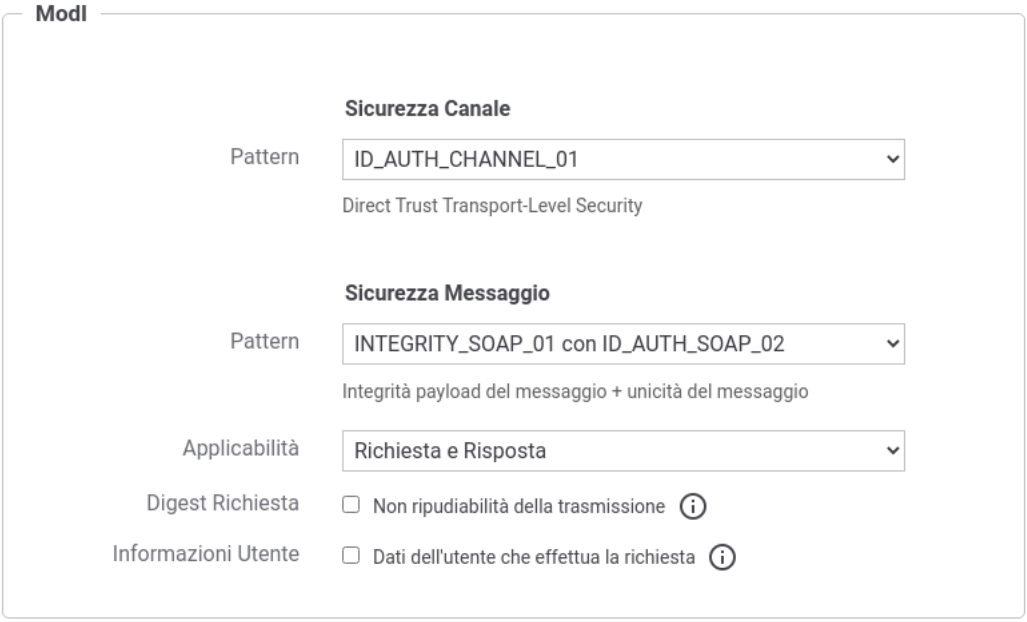


Fig. 3.27: Pattern di sicurezza messaggio «INTEGRITY_SOAP_01» per l'API

Occorre solo tenere presente che per questo pattern di sicurezza sono presenti le seguenti differenze sulle maschere di configurazione delle API di tipo REST:

- Nel contesto della configurazione di una fruizione, relativamente alla sezione «ModI - Richiesta», oltre ai dati da fornire per la produzione della firma digitale deve essere aggiunta anche l'indicazione degli eventuali Header HTTP da firmare. Tale indicazione viene fornita con il campo «HTTP Headers da firmare» (Fig. 3.28).

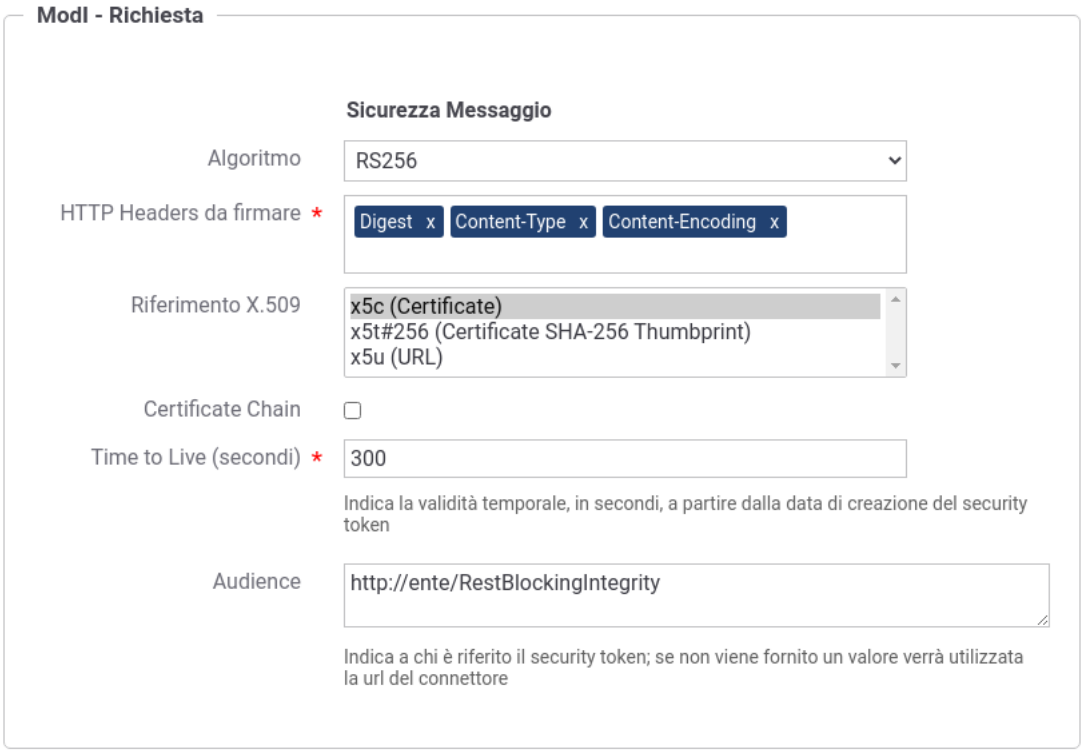


Fig. 3.28: Fruizione «INTEGRITY_REST_01» - Configurazione richiesta con indicazione HTTP Headers da firmare

- Nel contesto della configurazione di una erogazione, relativamente alla sezione «ModI - Risposta», oltre ai dati da fornire per la produzione della firma digitale deve essere aggiunta anche l'indicazione degli eventuali Header HTTP da firmare. Tale indicazione viene fornita con il campo «HTTP Headers da firmare» (Fig. 3.29).

3.3.5 Digest della Richiesta - Non ripudiabilità della trasmissione

Questa funzionalità consente di estendere «INTEGRITY_REST_01» aggiungendo all'interno del token di sicurezza della risposta il digest della richiesta.

La funzionalità consente di implementare la soluzione per la non ripudiabilità della trasmissione come suggerito nelle linee guida di interoperabilità (Fig. 3.30) all'interno del documento "03 Profili di interoperabilità.pdf".

Nota: La sigla che identifica il pattern di sicurezza messaggio varia a seconda se l'API sia di tipo REST, per cui la sigla corrisponde a *INTEGRITY_REST_01*, o SOAP dove viene utilizzata la sigla *INTEGRITY_SOAP_01*.

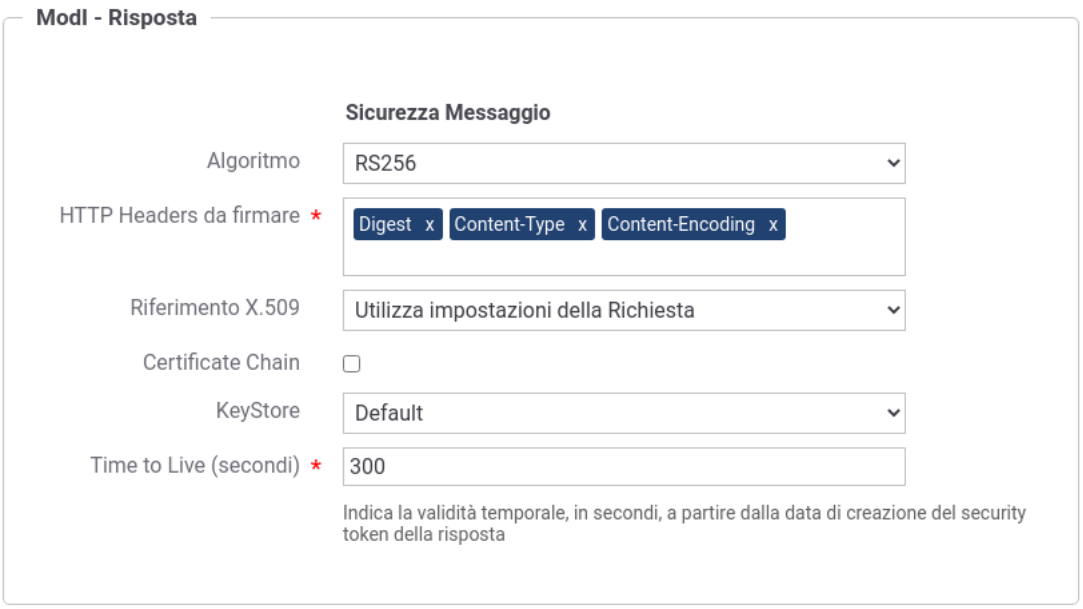


Fig. 3.29: Erogazione «INTEGRITY_REST_01» - Configurazione risposta con indicazione HTTP Headers da firmare

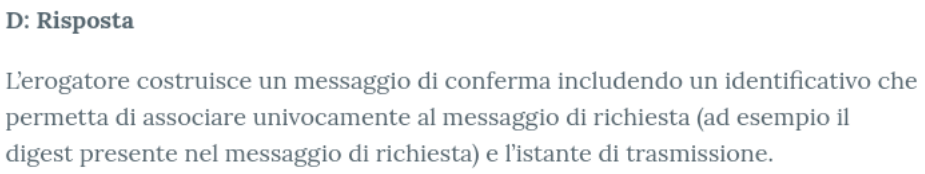


Fig. 3.30: Punto "D" della soluzione di sicurezza per la non ripudiabilità della trasmissione

L'attivazione di questo profilo avviene a livello della relativa API, nella sezione «ModI», elemento «Sicurezza Messaggio», selezionando la voce «Digest Richiesta» (Fig. 3.31).

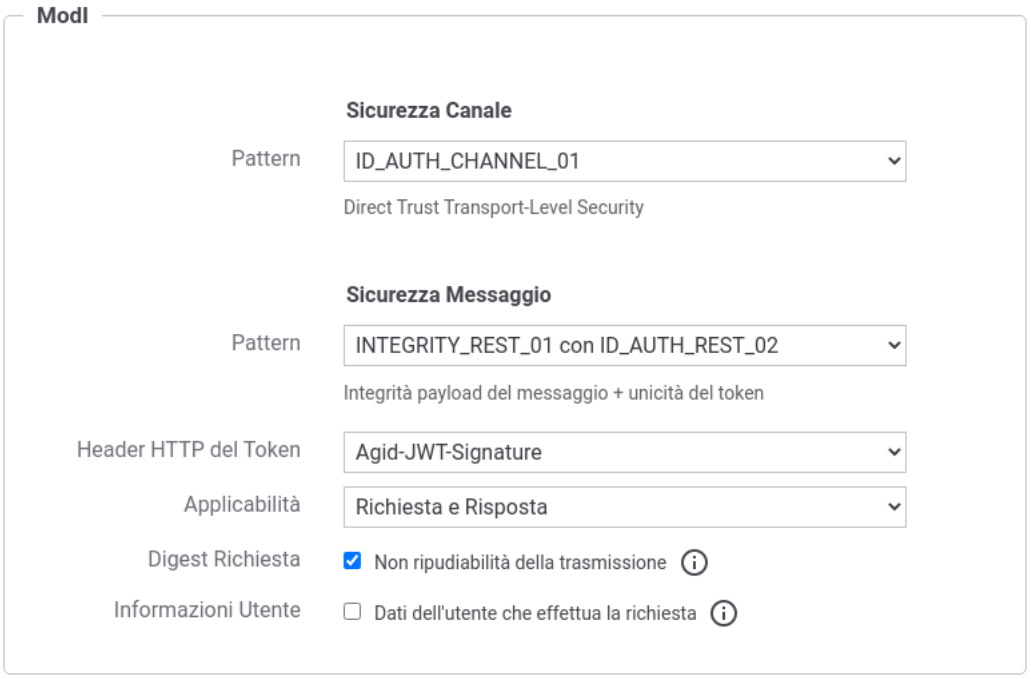


Fig. 3.31: Pattern di sicurezza messaggio «INTEGRITY_REST_01» + Digest Richiesta

Nota: Poichè la funzionalità è una estensione di «INTEGRITY_REST_01», la voce "Digest Richiesta" compare solamente se è stato selezionato il pattern «INTEGRITY_REST_01 con ID_AUTH_REST_01» o «INTEGRITY_REST_01 con ID_AUTH_REST_02»

Nota: Nel caso venga disabilitata la generazione della sicurezza messaggio sulla richiesta o sulla risposta, la funzionalità "Digest della Richiesta" non sarà più attivabile.

Nella figura Fig. 3.32 viene riportato un esempio del payload, relativo al token di sicurezza ModI della risposta per una API REST, contenente il digest della richiesta.

Nella figura Fig. 3.33 viene riportato un esempio relativo al token di sicurezza ModI della risposta per una API SOAP. Tutti i digest degli elementi firmati nella richiesta vengono riportati all'interno di un header soap "X-Digest-Richiesta" della risposta. Il nuovo header "X-Digest-Richiesta" sarà aggiunto agli elementi firmati nella risposta.

Fig. 3.32: Payload del Token di Sicurezza REST con pattern «INTEGRITY_REST_01» + Digest Richiesta

```
                <wsu:Expires>2020-06-23T09:51:40.499Z</wsu:Expires>
            </wsu:Timestamp>
        </wsse:Security>
        <ns2:X-RequestDigest xmlns:ns2="http://amministrazioneesempio.it/nomeinterfacciaservizio"
xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd" env:mustUnderstand="false" wsu:Id="id-fc810b92-431a-4f5f-a917-df88f173472d">
            <ds:Reference xmlns:ds="http://www.w3.org/2000/09/xmldsig#" URI="#TS-1d254b27-62ab-4798-bcd7-1b636a7af6f6">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                        <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="wsse soap"/>
                    </ds:Transform>
                </ds:Transforms>
                <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                <ds:DigestValue>mNF7nyQtYMEh9r28c5I0dzHt+G6xQsna1BG8Nl+KKxw=</ds:DigestValue>
            </ds:Reference>
            <ds:Reference xmlns:ds="http://www.w3.org/2000/09/xmldsig#" URI="#id-59659b84-bbe8-401c-bba0-25c731086b4b">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                        <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap"/>
                    </ds:Transform>
                </ds:Transforms>
                <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                <ds:DigestValue>Bz5zFFxlzesguIinBsugODk+URQTOKLeSIs+Uj8Fap4=</ds:DigestValue>
            </ds:Reference>
            <ds:Reference xmlns:ds="http://www.w3.org/2000/09/xmldsig#" URI="#id-8bce48ae-9c13-40ae-b888-8e3a6131b71c">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                        <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap"/>
                    </ds:Transform>
```

Fig. 3.33: Payload del Token di Sicurezza SOAP con pattern «INTEGRITY_SOAP_01» + Digest Richiesta

3.3.6 Informazioni Utente

Questa funzionalità consente di estendere «INTEGRITY_REST_01» aggiungendo all'interno del token di sicurezza le informazioni sull'utente che ha effettuato la richiesta.

Nota: La sigla che identifica il pattern di sicurezza messaggio varia a seconda se l'API sia di tipo REST, per cui la sigla corrisponde a *INTEGRITY_REST_01*, o SOAP dove viene utilizzata la sigla *INTEGRITY_SOAP_01*.

L'attivazione di questa funzionalità avviene a livello della relativa API, nella sezione «ModI», elemento «Sicurezza Messaggio», selezionando la voce «Informazioni Utente» (Fig. 3.34).

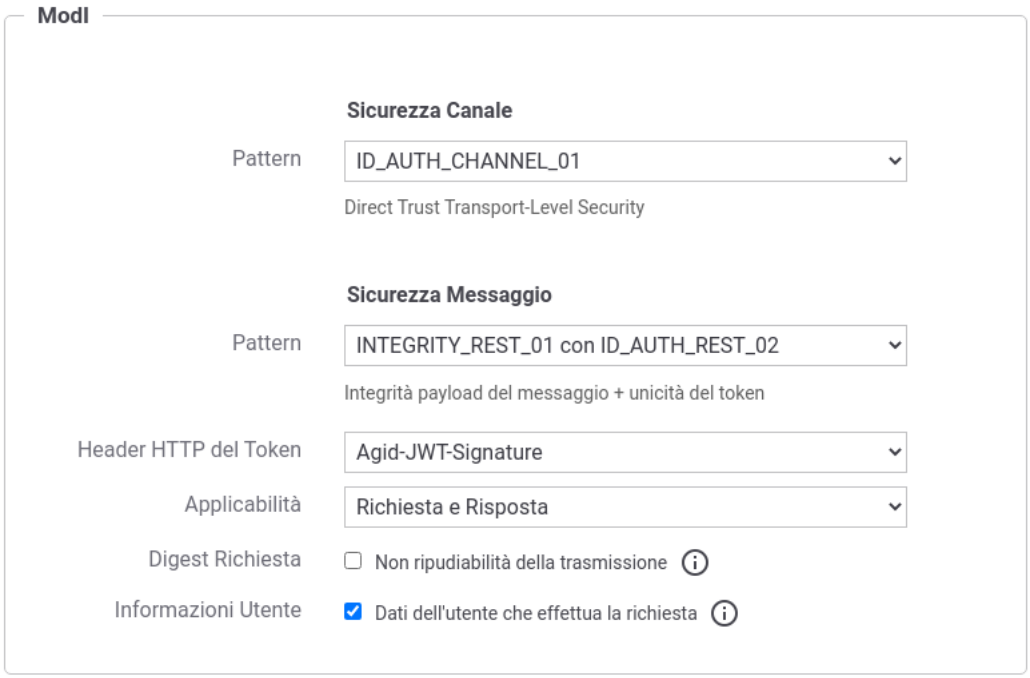


Fig. 3.34: Pattern di sicurezza messaggio «INTEGRITY_REST_01» + Informazioni Utente

---

Nota: Poichè la funzionalità è una estensione di «INTEGRITY_REST_01», la voce "Informazioni Utente" compare solamente se è stato selezionato il pattern «INTEGRITY_REST_01 con ID_AUTH_REST_01» o «INTEGRITY_REST_01 con ID_AUTH_REST_02»

Nota: Nel caso venga disabilitata la generazione della sicurezza messaggio sulla richiesta, la funzionalità "Informazioni Utente" non sarà più attivabile.

Le informazioni aggiuntive presenti all'interno del token riguardano:

- UserID Utente: identificativo univoco dell'utente all'interno del dominio rappresentato dal "Codice Ente";
- Indirizzo IP Utente: identifica la postazione da cui l'utente ha effettuato la richiesta;
- Codice Ente: dominio di appartenenza dell'utente.

Nella figura Fig. 3.35 viene riportato un esempio del payload relativo al token di sicurezza "ModI" di una API REST, contenente le informazioni aggiuntive sull'utente che ha effettuato la richiesta.

Fig. 3.35: Payload del Token di Sicurezza REST con pattern «INTEGRITY_REST_01» + Informazioni Utente

Nella figura Fig. 3.36 viene riportato un esempio relativo al token di sicurezza "ModI" per una API SOAP. Le informazioni aggiuntive sull'utente che ha effettuato la richiesta sono incluse in una Asserziona SAML.

In una fruizione, le informazioni aggiuntive che vengono aggiunte nel token, sono per default attese nella richiesta pervenuta a GovWay sotto forma di header http o parametro della url:

- UserID Utente: l'identificativo dell'utente deve essere indicato nella richiesta di fruizione all'interno dell'header http "GovWay-CS-User" o del parametro della url con nome "govway_cs_user";
- Indirizzo IP Utente: la postazione dell'utente deve essere indicata nell'header http "GovWay-CS-IPUser" o del parametro della url con nome "govway_cs_ipuser";
- Codice Ente: per default questa informazione assume il valore del soggetto registrato su GovWay, di dominio interno, per il quale si sta effettuando la richiesta di fruizione dell'API.

Il comportamento di default, per l'acquisizione dei valori utilizzati per le tre informazioni aggiuntive, può essere personalizzato accedendo nella sezione «ModI» di una fruizione, e modificando le voci «Informazioni Utente» (Fig. 3.37) indicando un valore statico o utilizzando le proprietà dinamiche descritte nella sezione *Valori dinamici*.

```
            </wsu:Timestamp>
            <saml2:Assertion xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" ID="_4d961d77-fac5-4f9c-a5a6-3fb7506b058f" IssueInstant="2020-06-23T09:46:40.112Z" Version="2.0"
xsi:type="saml2:AssertionType">
                <saml2:Issuer Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">EnteFruitore</saml2:Issuer>
                <saml2:Subject>
                    <saml2:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">EnteFruitore</saml2:NameID>
                    <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:sender-vouches">
                        <saml2:SubjectConfirmationData NotBefore="2020-06-23T09:46:40.112Z" NotOnOrAfter="2020-06-23T10:46:40.112Z"/>
                    </saml2:SubjectConfirmation>
                </saml2:Subject>
                <saml2:Conditions NotBefore="2020-06-23T09:46:40.112Z" NotOnOrAfter="2020-06-23T10:46:40.112Z"/>
                <saml2:AuthnStatement AuthnInstant="2020-06-23T09:46:40.112Z" SessionNotOnOrAfter="2020-06-23T10:46:40.112Z">
                    <saml2:AuthnContext>
                        <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:unspecified</saml2:AuthnContextClassRef>
                    </saml2:AuthnContext>
                </saml2:AuthnStatement>
                <saml2:AttributeStatement>
                    <saml2:Attribute Name="User" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified">
                        <saml2:AttributeValue xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:type="xsd:string">mariorossi</saml2:AttributeValue>
                    </saml2:Attribute>
                    <saml2:Attribute Name="IP-User" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:unspecified">
                        <saml2:AttributeValue xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xsi:type="xsd:string">10.114.87.24</saml2:AttributeValue>
                    </saml2:Attribute>
                </saml2:AttributeStatement>
            </saml2:Assertion>
        </wsse:Security>
        <wsa:To xmlns:wsa="http://www.w3.org/2005/08/addressing" xmlns:env="http://www.w3.org/2003/05/soap-envelope"
```

Fig. 3.36: Payload del Token di Sicurezza SOAP con pattern «INTEGRITY_SOAP_01» + Informazioni Utente

3.3.7 Funzionalità Avanzate

La gestione dei pattern di sicurezza messaggio possono essere personalizzati su diversi aspetti:

- *Attivazione della sicurezza messaggio su richiesta/risposta*: è possibile attivare la sicurezza messaggio puntualmente solamente sulla richiesta o sulla risposta di una operazione. Per API REST è possibile anche definire dei criteri di applicabilità della sicurezza messaggio in base a Content-Type o codici di risposta HTTP.
- *Header HTTP del token JWT*: può essere selezionato l'header http utilizzato per veicolare il token JWT su API REST.
- *Payload Claims del token JWT*: possono essere configurati ulteriori claims da aggiungere nel payload del JWT su API REST.
- *Header SOAP aggiunti nella WSSecurity Signature*: possono essere configurati ulteriori header soap da aggiungere agli elementi inclusi nella firma su API SOAP.
- *Eliminazione token/header contenente la sicurezza messaggio*: è possibile configurare GovWay al fine di non eliminare il token di sicurezza dai messaggi dopo averli validati.

Attivazione della sicurezza messaggio su richiesta/risposta

Insieme all'attivazione di un pattern di sicurezza messaggio è possibile configurarne l'attivazione solamente sulla richiesta o sulla risposta (Fig. 3.38).

Per API REST è possibile anche definire dei criteri di applicabilità della sicurezza messaggio in base a Content-Type o codici di risposta HTTP selezionando la voce "Personalizza criteri di applicabilità". La personalizzazione dei criteri consente di differenziare la configurazione tra richiesta e risposta come mostrato nella figura Fig. 3.39:

- Richiesta: oltre ad abilitare o disabilitare, è consentito definire una lista di Content-Type solamente per i quali verrà attuata la sicurezza messaggio sulla richiesta.

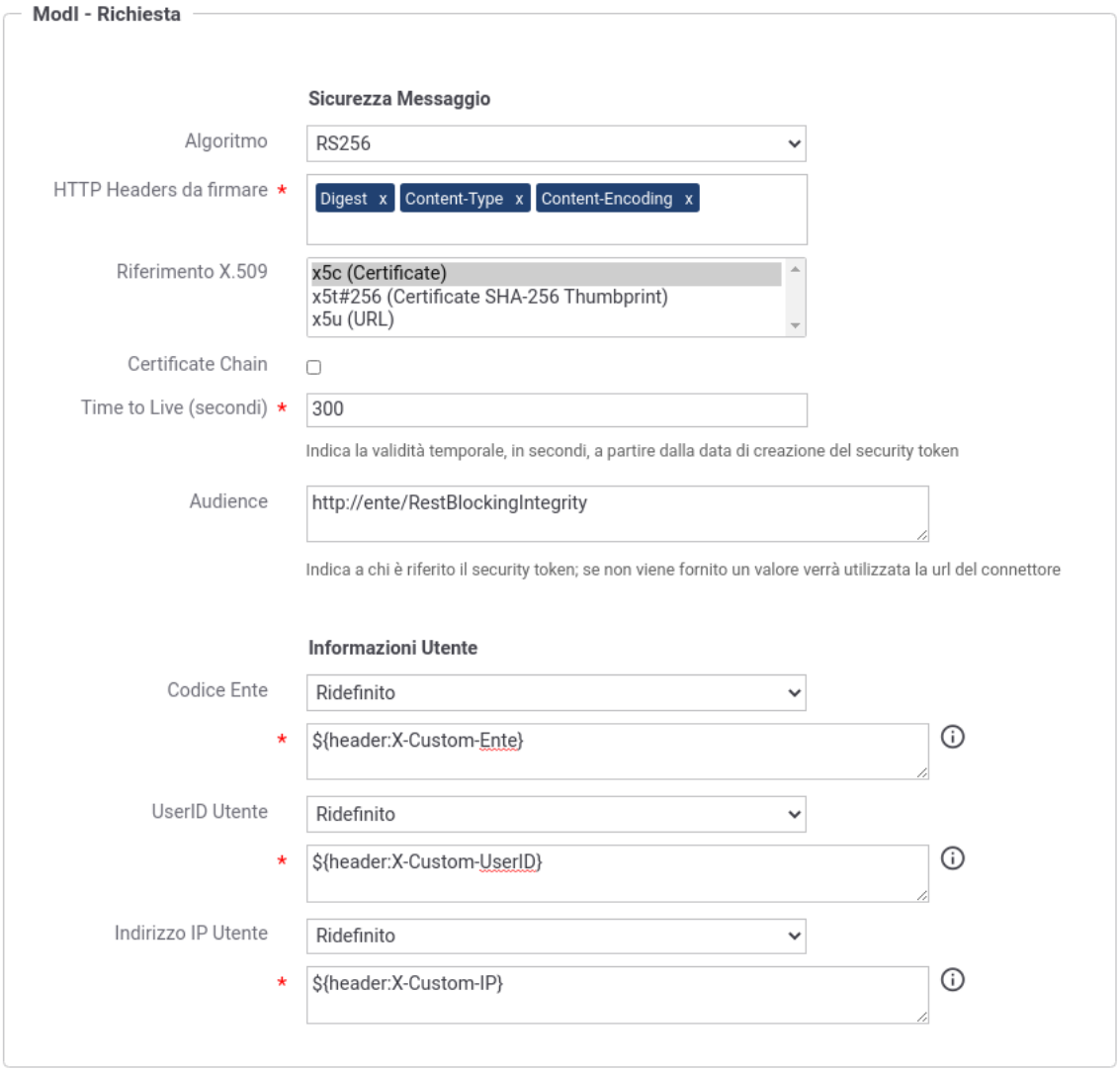


Fig. 3.37: Personalizzazione dell'acquisizione delle Informazioni Utente

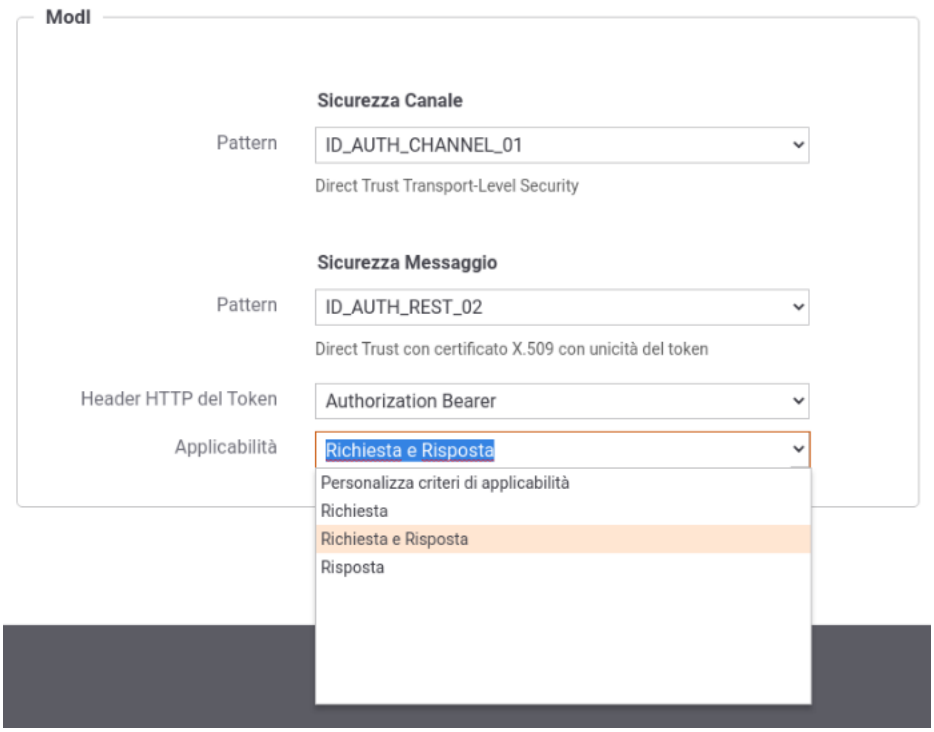


Fig. 3.38: Configurazione dell'applicabilità della sicurezza messaggio

- Risposta: oltre ad abilitare o disabilitare, è consentito definire una lista di Content-Type e/o una lista di codice di risposta HTTP per i quali verrà attuata la sicurezza messaggio sulla risposta.

La lista di Content-Type per i quali la sicurezza messaggio verrà utilizzata è definibile tramite i seguenti formati:

- type/subtype: indicazione puntuale di un Content-Type
- type/*: hanno un match tutti i Content-Type appartenenti al tipo indicato
- */*+xml: hanno un match tutti i Content-Type che terminano con "+xml"
- regexpType/regexpSubType: hanno un match tutti i Content-Type che soddisfano le espressioni regolari indicate
- empty: valore speciale che rappresenta una richiesta senza Content-Type

La lista dei codici di risposta HTTP per i quali la sicurezza messaggio verrà utilizzata può contenere un codice http puntuale (es. 200) o un intervallo fornendo due codici separati dal trattino (es. 200-299).

Header HTTP del token JWT

Il pattern di sicurezza, su API di tipo REST, produrrà la generazione di un token JWT firmato inserito all'interno dell'header HTTP previsto dalle *Linee Guida AGID di Interoperabilità* (LG) dove vengono definiti gli header HTTP "Authorization" (Bearer) da usare per l'auenticazione e l'header HTTP "Agid-JWT-Signature" per l'integrità.

La configurazione di default degli header prodotti varia a seconda del pattern di sicurezza selezionato:

- *[ID_AUTH_SOAP_01 / ID_AUTH_REST_01] Direct Trust con certificato X.509*: header HTTP "Authorization".
- *[ID_AUTH_SOAP_02 / ID_AUTH_REST_02] Direct Trust con certificato X.509 con unicità del messaggio/token*: header HTTP "Authorization".

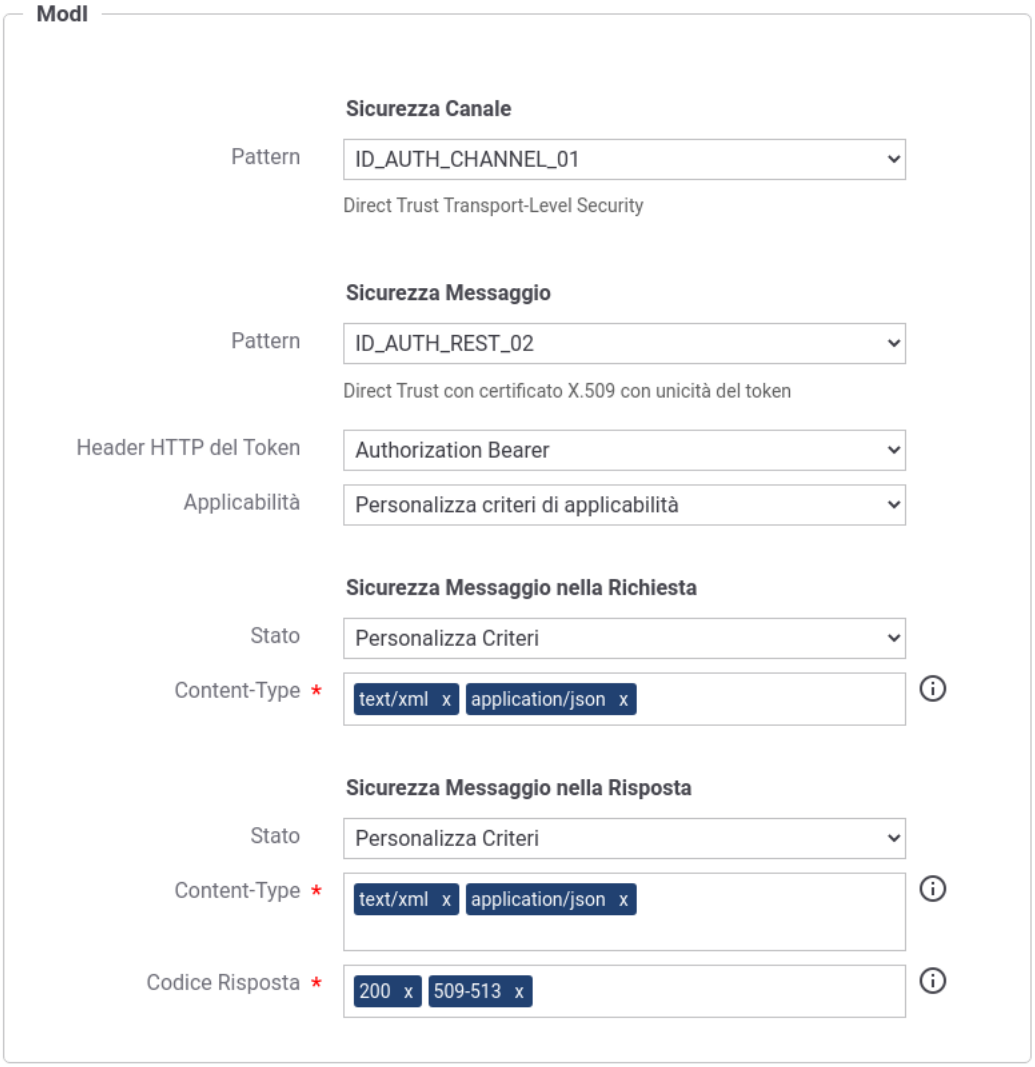


Fig. 3.39: Configurazione dell'applicabilità della sicurezza messaggio personalizzata per Content-Type e Codici di Risposta

- *[INTEGRITY_SOAP_01 / INTEGRITY_REST_01] Integrità payload del messaggio*: nel flusso di richiesta vengono prodotti entrambi gli header, mentre nel flusso di risposta solamente l'header HTTP "Agid-JWT-Signature".

La voce "Header HTTP del Token" consente di modificare la configurazione di default e variare sia il nome che l'eventuale contemporaneità dei 2 header principalmente per due motivi:

- per consentire la retrocompatibilità con il pattern IDAR03, previsto nelle linee guida della versione "bozza" (https://docs.italia.it/italia/piano-triennale-ict/lg-modellointeroperabilita-docs/it/bozza/), dove veniva utilizzato con qualsiasi pattern sempre un unico header HTTP "Authorization";
- per supportare qualsiasi interpretazione del pattern "INTEGRITY_REST_01" e relativa implementazione da parte della controparte con cui si deve interoperare.

Per i motivi suddetti le possibili configurazioni supportate configurabili tramite la voce "Header HTTP del Token" sono le seguenti (Fig. 3.40):

- "Agid-JWT-Signature + Authorization Bearer" : opzione selezionabile solamente con pattern che prevede "INTEGRITY_REST_01". Prevede la generazione nel flusso di richiesta di entrambi gli header http previsti dalle LG. Nel flusso di risposta è previsto invece solamente l'header "Agid-JWT-Signature".
- "Agid-JWT-Signature + Authorization Bearer anche nella risposta": comportamento identico all'opzione precedente, dove però la contemporaneità dei due header è prevista anche nel flusso di risposta.
- "Agid-JWT-Signature" : viene generato sempre e solo un unico token di sicurezza, indipendentemente dal pattern di sicurezza selezionato, utilizzando come nome dell'header HTTP il nome proprietario delle LG.
- "Authorization Bearer" : comportamento identico all'opzione precedente, dove però viene utilizzato l'header HTTP "Authorization" e prefisso "Bearer " nel valore. Questa opzione, in presenza di pattern che prevede "INTEGRITY_REST_01", consente di essere interoperabile con i servizi implementati con la versione "bozza" delle LG.

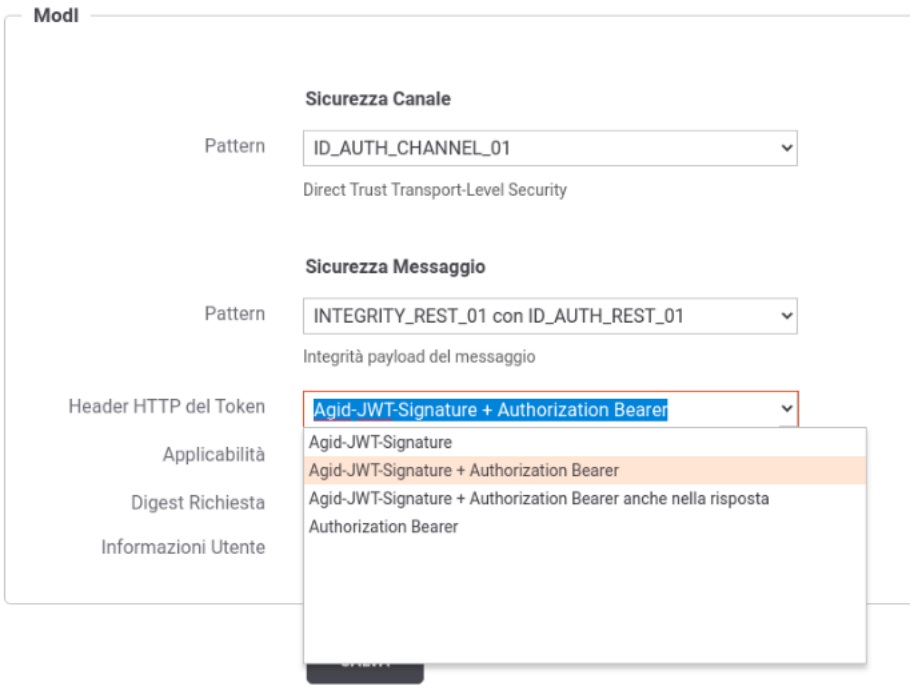


Fig. 3.40: Selezione dell'Header HTTP del token JWT

Nota: Processamento dell'header Agid-JWT-Signature con opzioni che prevedono anche l'header Authorization

Se un'API o una risorsa è stata configurata con un'opzione che prevede la contemporaneità dei 2 header, la generazione o la verifica dell'header "Agid-JWT-Signature" avviene solamente se la richiesta o la risposta prevede un payload; ad esempio in una richiesta HTTP GET non verrà generato o atteso l'header. Caso eccezionale riguarda i flussi configurati per firmare header http indipendenti dal payload della richiesta (diversi quindi dai classici header Content-Type, Content-Encoding, Digest); in questi casi l'header "Agid-JWT-Signature" verrà generato anche in assenza di payload poichè nel token saranno inseriti nel claim "signed_headers" gli ulteriori header configurati per essere firmati.

Se in un'API viene selezionata una opzione che prevede la contemporaneità dei 2 header, nelle maschere di configurazione ModI delle fruizioni e delle erogazioni saranno presenti ulteriori opzioni che consentono di personalizzare la gestione dei claims presenti all'interno dei due header. Di seguito vengono descritte le opzioni ulteriori presenti nelle fruizioni e nelle erogazioni

Configurazione contemporaneità degli header in una Fruizione

La configurazione differisce a seconda se il token di sicurezza deve essere generato (Richiesta) o validato (Risposta). Di seguito vengono descritte le opzioni di configurazione possibili per le due fasi.

Richiesta

Di seguito vengono descritti i parametri di configurazione riguardanti la generazione dei due token nel flusso di richiesta.

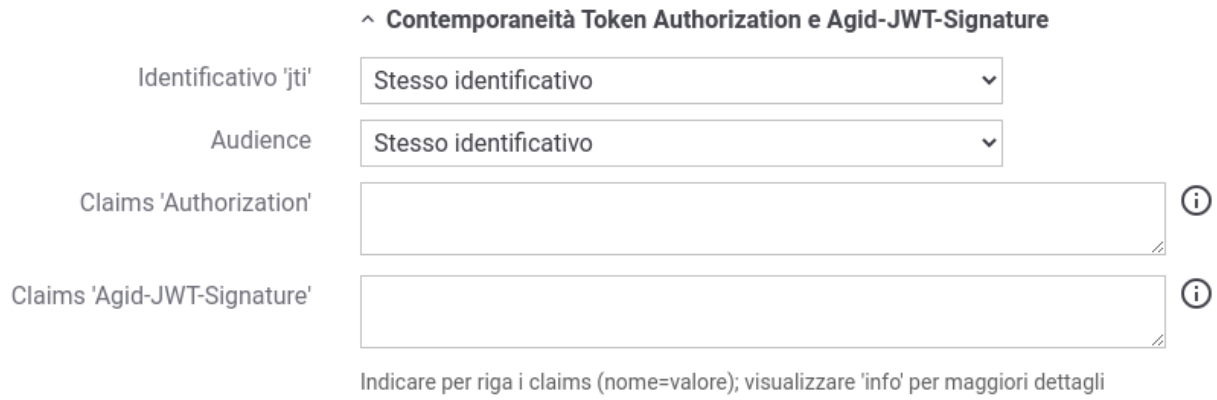


Fig. 3.41: Maschera ModI per la configurazione della fruizione in presenza dei due header nella richiesta

- Identificativo "jti": la configurazione di default prevede la valorizzazone dl claim "jti", presente all'interno dei token di sicurezza portati dai due header, con il medesimo identificativo. Il parametro consente di impostare la generazione di un identificativo differente e di indicare al gateway quale dei due identificativi dovrà associare alla traccia come "ID del Messaggio" tramite la voce "Usa come ID Messaggio" (Fig. 3.42). L'identificativo indicato sarà utilizzabile come criterio di ricerca puntuale tramite la funzionalità disponibile con la console e le API di monitoraggio. L'identificativo "jti" presente nell'altro header verrà comunque associato alla traccia ma non sarà direttamente utilizzabile come criterio di ricerca per individuare la transazione in un secondo momento.
- Audience: la configurazione di default prevede la valorizzazione del claim "aud", presente all'interno dei token di sicurezza portati dai due header, con il medesimo identificativo. Il parametro consente di im-

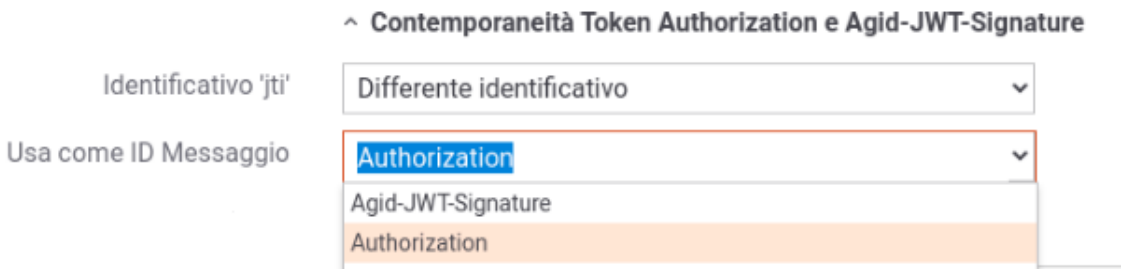


Fig. 3.42: Selezione del claim "jti" da utilizzare come ID del Messaggio

postare la generazione di un identificativo differente indicando l'audience da impostare nel token dell'header Agid-JWT-Signature come mostrato nella figura Fig. 3.43.

Fig. 3.43: Configurazione del claim "aud" da utilizzare nel token dell'header Agid-JWT-Signature

- Claims "Authorization" e "Agid-JWT-Signature": consente di modificare i valori di default associati ai claim standard e/o di definirne altri solamente all'interno del token dell'header indicato. Per maggiori dettagli sulle configurazioni disponibili si rimanda alla sezione *Payload Claims del token JWT*.

Risposta

Di seguito vengono descritti i parametri di configurazione riguardanti la validazione dei due token nel flusso di risposta.

Fig. 3.44: Maschera ModI per la configurazione della fruizione in presenza dei due header nella risposta

- Id "jti" per Filtro Duplicati: consente di indicare da quale header estrarre l'identificativo "jti" da associare alla traccia come "ID del Messaggio" (default: Agid-JWT-Signature). L'identificativo indicato verrà utilizzato per la funzionalità di filtro delle richieste duplicate. Inoltre l'identificativo indicato sarà utilizzabile come criterio di ricerca puntuale tramite la funzionalità disponibile con la console e le API di monitoraggio. L'identificativo "jti" presente nell'altro header verrà comunque associato alla traccia ma non sarà direttamente utilizzabile come criterio di ricerca per individuare la transazione in un secondo momento.
- Audience: la configurazione di default si attende un valore del claim "aud", presente all'interno dei token di sicurezza portati dai due header, con il medesimo identificativo. Il parametro consente di impostare la generazione di un identificativo differente indicando l'audience atteso nell'header Agid-JWT-Signature come mostrato nella figura Fig. 3.45.

Fig. 3.45: Valore atteso per il claim "aud" nel token dell'header Agid-JWT-Signature

Configurazione contemporaneità degli header in una Erogazione

La configurazione differisce a seconda se il token di sicurezza deve essere validato (Richiesta) o generato (Risposta). Di seguito vengono descritte le opzioni di configurazione possibili per le due fasi.

Richiesta

Di seguito vengono descritti i parametri di configurazione riguardanti la validazione dei due token nel flusso di richiesta.

Fig. 3.46: Maschera ModI per la configurazione dell'erogazione in presenza dei due header nella richiesta

- Id "jti" per Filtro Duplicati: consente di indicare da quale header estrarre l'identificativo "jti" da associare alla traccia come "ID del Messaggio" (default: Agid-JWT-Signature). L'identificativo indicato verrà utilizzato per la funzionalità di filtro delle richieste duplicate. Inoltre l'identificativo indicato sarà utilizzabile come criterio di ricerca puntuale tramite la funzionalità disponibile con la console e le API di monitoraggio. L'identificativo "jti" presente nell'altro header verrà comunque associato alla traccia ma non sarà direttamente utilizzabile come criterio di ricerca per individuare la transazione in un secondo momento.
- Audience: la configurazione di default si attende un valore del claim "aud", presente all'interno dei token di sicurezza portati dai due header, con il medesimo identificativo. Il parametro consente di impostare la generazione di un identificativo differente indicando l'audience atteso nell'header Agid-JWT-Signature come mostrato nella figura Fig. 3.45.

Risposta

Di seguito vengono descritti i parametri di configurazione riguardanti la generazione dei due token nel flusso di risposta.

- Identificativo "jti": la configurazione di default prevede la valorizzazone dl claim "jti", presente all'interno dei token di sicurezza portati dai due header, con il medesimo identificativo. Il parametro consente di impostare la generazione di un identificativo differente e di indicare al gateway quale dei due identificativi dovrà associare alla traccia come "ID del Messaggio" tramite la voce "Usa come ID Messaggio" (Fig. 3.42). L'identificativo indicato sarà utilizzabile come criterio di ricerca puntuale tramite la funzionalità disponibile con la console e le

Fig. 3.47: Maschera ModI per la configurazione dell'erogazione in presenza dei due header nella risposta

API di monitoraggio. L'identificativo "jti" presente nell'altro header verrà comunque associato alla traccia ma non sarà direttamente utilizzabile come criterio di ricerca per individuare la transazione in un secondo momento.

- Claims "Authorization" e "Agid-JWT-Signature": consente di modificare i valori di default associati ai claim standard e/o di definirne altri solamente all'interno del token dell'header indicato. Per maggiori dettagli sulle configurazioni disponibili si rimanda alla sezione *Payload Claims del token JWT*.

Payload Claims del token JWT

Il pattern di sicurezza, su API di tipo REST, produrrà la generazione di un token JWT firmato inserito all'interno dell'header HTTP previsto dalle *Linee Guida AGID di Interoperabilità*. Nel payload del JWT vengono generati i claim di default previsti dal prodotto come quelli temporali (iat, nbf, exp), l'identificativo unico della richiesta (jti), e altri claims che consentono di individuare gli attori (sub, iss, client_id, aud).

Altri claims possono essere aggiunti al payload JWT definendoli nel campo "Claims" tra i criteri di configurazione "ModI" della richiesta, in una fruizione, o della risposta, in una erogazione. Vanno indicati per riga nella forma "nome=valore" come mostrato nella figura Fig. 3.48. Il valore può essere definito come costante o contenere parti dinamiche, definite tramite una sintassi proprietaria di GovWay, che verranno risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*).

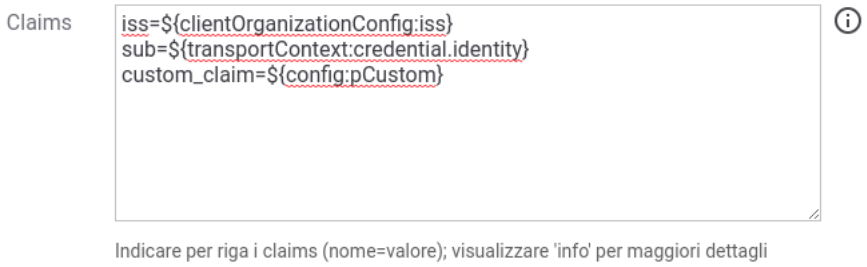


Fig. 3.48: Claims aggiuntivi inseriti nel Payload JWT

Nota: Non è consentito indicare i claims "iat, nbf, exp, jti". In una richiesta non è inoltre consentito indicare ne il claim "aud" ne il claim "client_id" (quest'ultimo prevede un caso eccezionale con il valore ${notGenerate} descritto in seguito). In una risposta, invece, non è consentito indicare il claim "request_digest".

Di seguito vengono forniti i valori di default inseriti da GovWay nel payload jwt per quanto concerne i claims che individuano gli attori, differenziando tra il token di richiesta generato da una fruizione e il token di risposta generato da una erogazione. Per ogni claim viene anche indicato come modificare il valore di default associato.

- "aud": indica a chi è riferito il security token
  - fruizione: il valore da inserire nel payload JWT può essere indicato tra i criteri di configurazione "ModI", nella sezione richiesta. Se non viene fornito un valore verrà utilizzata la url del connettore.
  - erogazione: viene utilizzato il primo valore che ha un match in ordine con i seguenti criteri:
    - claim "aud" indicato nel campo "Claims" tra i criteri di configurazione "ModI" della risposta;
    - valore configurato nel campo "Identificativo Client" dell'applicativo mittente identificato;
    - valore presente nel claim "client_id" del payload JWT ricevuto nella richiesta;
    - valore presente nel claim "sub" del payload JWT ricevuto nella richiesta;
    - valore "anonymous"
- "iss": identificativo del soggetto che ha rilasciato (e firmato) il token; viene utilizzato il primo valore che ha un match in ordine con i seguenti criteri:
  - claim "iss" indicato nel campo "Claims" tra i criteri di configurazione "ModI" della richiesta, in una fruizione, o della risposta, in una erogazione;
  - identificativo del soggetto fruitore in una fruizione o l'identificativo del soggetto erogatore in una erogazione
- "sub": identificativo del mittente a cui è riferito il token; viene utilizzato il primo valore che ha un match in ordine con i seguenti criteri:
  - claim "sub" indicato nel campo "Claims" tra i criteri di configurazione "ModI" della richiesta, in una fruizione, o della risposta, in una erogazione;
  - identificativo dell'applicativo mittente in una fruizione o l'identificativo e la versione dell'API implementata in una erogazione
- "client_id": identificativo dell'applicazione client che ha ottenuto il token; viene utilizzato il primo valore che ha un match in ordine con i seguenti criteri:
  - fruizione:
    - valore configurato nel campo "Identificativo Client" dell'applicativo mittente identificato;
    - identificativo dell'applicativo mittente
  - erogazione:
    - claim "client_id" indicato nel campo "Claims" tra i criteri di configurazione "ModI" della risposta;
    - identificativo e versione dell'API implementata

Nota: È possibile utilizzare la keyword "${notGenerate}" come valore dei claims "iss", "sub" o "client_id", indicati nel campo "Claims" tra i criteri di configurazione "ModI", per non far generare il claim all'interno del jwt payload.

Header SOAP aggiunti nella WSSecurity Signature

Il pattern di sicurezza, su API di tipo SOAP, richiede la creazione una WSSecurity Signature dove all'interno vengono firmati gli elementi principali della richiesta (Timestamp, wsa:To) e gli altri elementi richiesti dai profili *[ID_AUTH_SOAP_02 / ID_AUTH_REST_02] Direct Trust con certificato X.509 con unicità del messaggio/token* (wsa:MessageId) e *[INTEGRITY_SOAP_01 / INTEGRITY_REST_01] Integrità payload del messaggio* (Body).

È possibile aggiungere, tra gli elementi firmati, ulteriori header SOAP oltre a quelli previsti dalla specifica ModI. Gli ulteriori header possono essere indicati nell'elemento "SOAP Headers da firmare" presente nella sezione "ModI - Richiesta" di una fruizione o nella sezione "ModI - Risposta" di una erogazione, come mostrato nella figura Fig. 3.49. Gli header devono essere definiti su ogni riga tramite la sintassi:

- {namespace}localName

Ad esempio:

- {http://example.govway.org}NomeHeader1
- {http://example.govway.org}NomeHeader2

Nota: L'elemento "SOAP Headers da firmare" è disponibile solamente utilizzando la govwayConsole in modalità *avanzata* (sezione *Modalità Avanzata*).

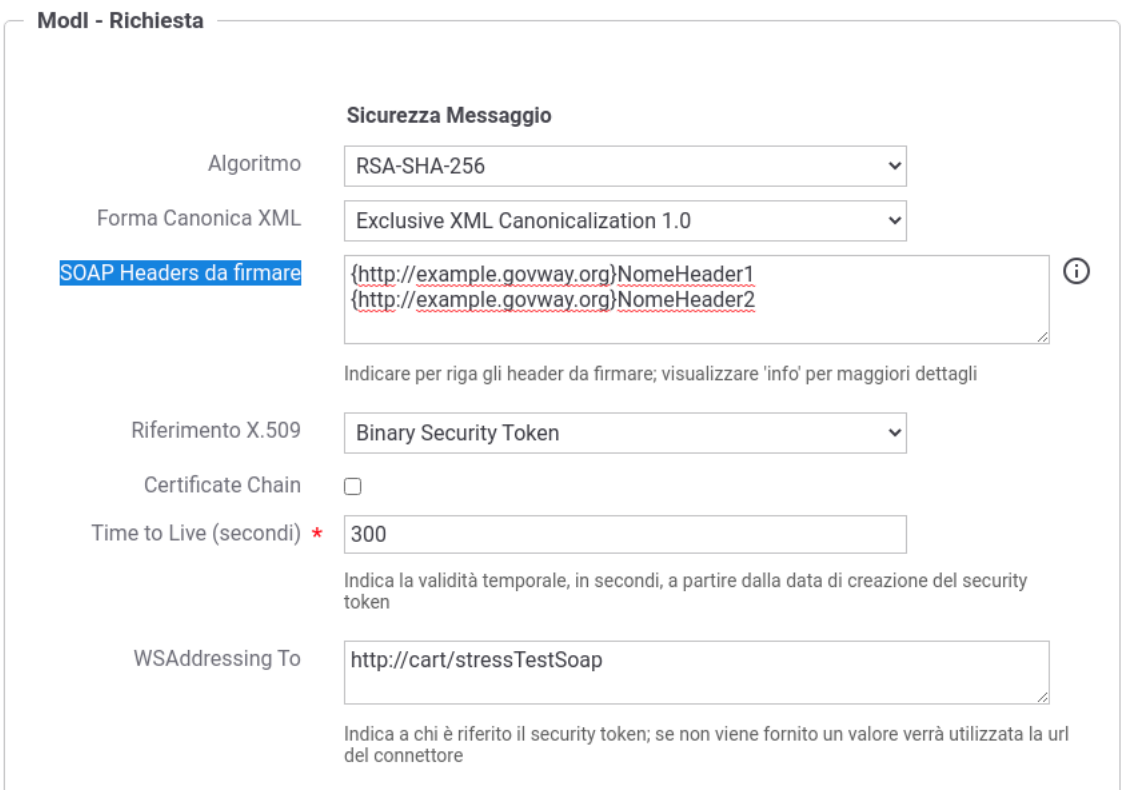


Fig. 3.49: Configurazione Header SOAP aggiuntivi da aggiungere alla firma

Eliminazione token/header contenente la sicurezza messaggio

I Token di sicurezza, dopo essere stati validati da GovWay, vengono eliminati dai messaggi in modo da rendere trasparente agli applicativi la gestione della sicurezza che è stata effettata sul Gateway.

È possibile, se necessario, configurare GovWay al fine di non fargli eliminare il token di sicurezza dai messaggi dopo averli validati. Per farlo si deve utilizzare la govwayConsole in modalità avanzata (vedi sezione *Modalità Avanzata*).

Per quanto concerne le richieste inoltrate ad un backend, durante la gestione di una erogazione, è possibile disabilitare l'eliminazione del token di sicurezza intervenendo sul connettore dell'erogazione e disabilitando la voce "Sbustamento ModI" all'interno della sezione "Trattamento Messaggio " come mostrato nella figura Fig. 3.50.

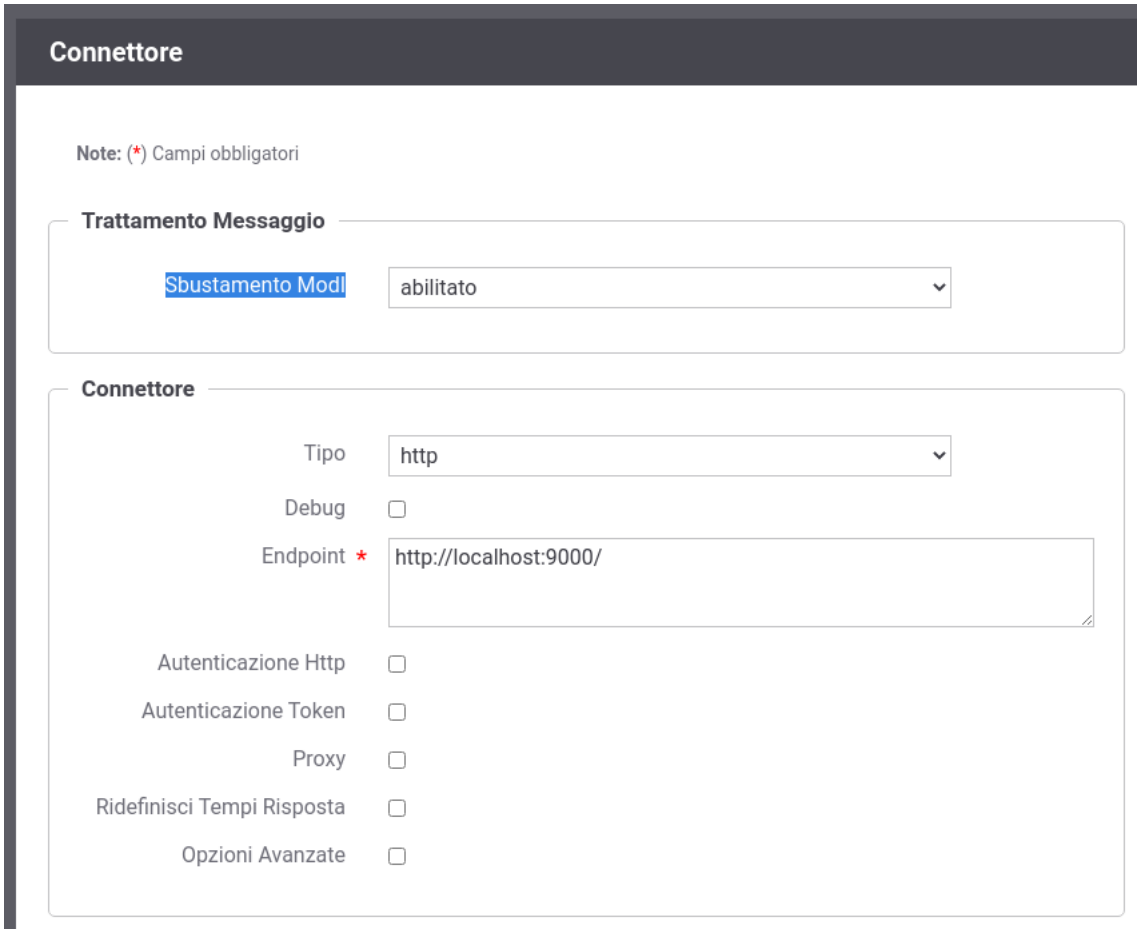


Fig. 3.50: Funzionalità "Sbustamento ModI" disabilitata per la Richiesta

Sulle risposte ritornate all'applicativo mittente, durante la gestione di una fruizione, è possibile disabilitare l'eliminazione del token di sicurezza intervenendo sull'applicativo e disabilitando la voce "Sbustamento ModI" all'interno della sezione "Trattamento Messaggio " come mostrato nella figura Fig. 3.51.

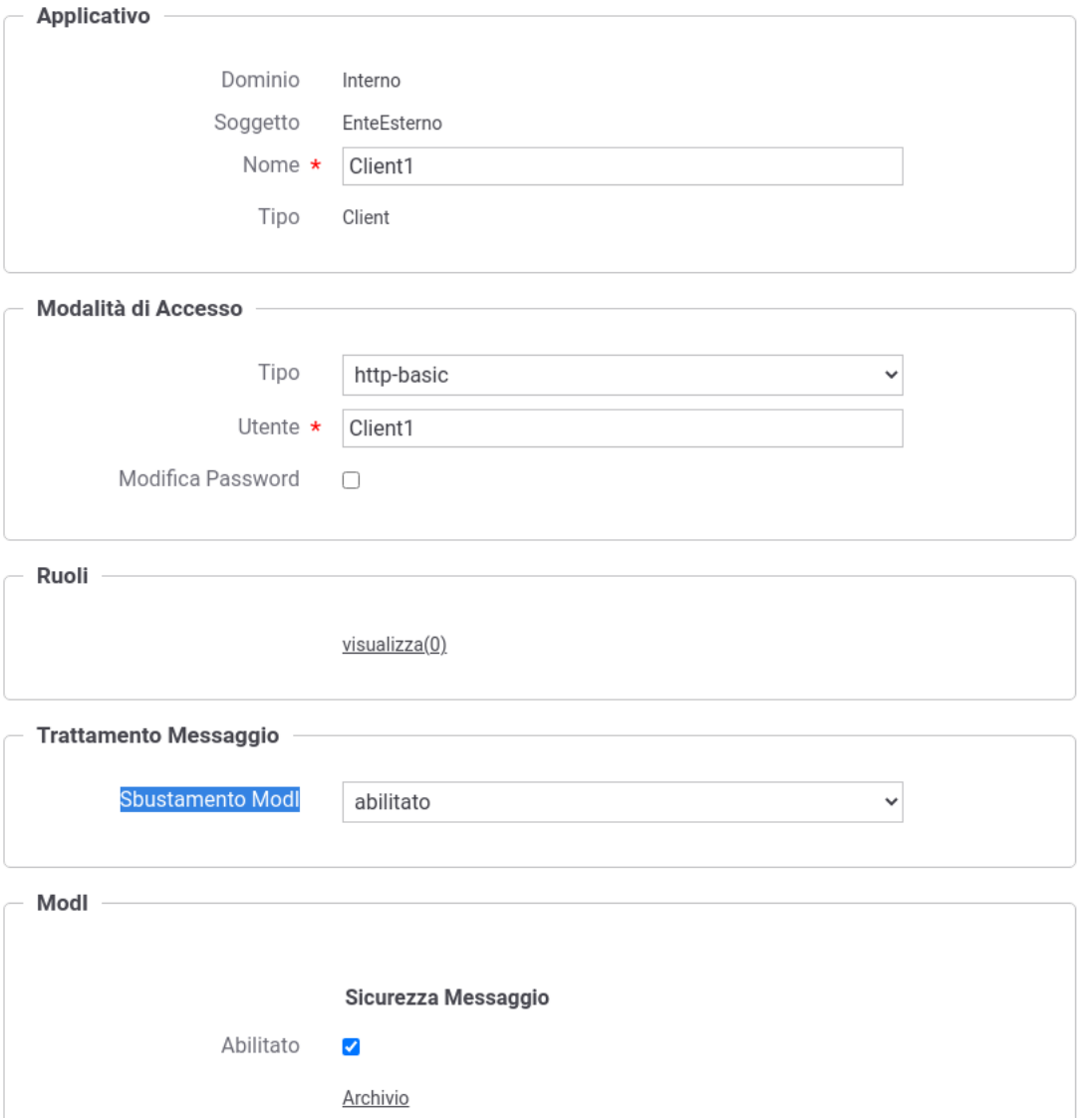


Fig. 3.51: Funzionalità “Sbustamento ModI” disabilitata per la Risposta

3.4 Pattern di Interazione

Le specifiche del Modello di Interoperabilità definiscono i Pattern di Interazione come le modalità secondo le quali un erogatore e un fruitore possono interagire. La distinzione operata a livello della specifica è quella tra il pattern «Bloccante» e quello «Non Bloccante». Solamente per API di tipo REST è disponibile anche un terzo pattern «Accesso CRUD» orientato alle risorse dove le API vengono utilizzate per eseguire operazioni di tipo CRUD - Create, Read, Update, Delete su risorse del dominio di interesse. Per le differenze di dettaglio tra i pattern si rimanda al testo della specifica.

Il pattern di interazione viene definito nell'interfaccia del servizio e conseguentemente GovWay recepisce tale informazioni nell'ambito della configurazione di una API nel contesto del profilo ModI.

La configurazione di API con il profilo ModI produce per default servizi con pattern di interazione «Bloccante» su API di tipo SOAP e «Accesso CRUD» su API di tipo REST. Se si desidera, è possibile modificare questa impostazione intervenendo puntualmente sulle singole operation/risorse della API.

La maschera di editing della singola operation/risorsa possiede la sezione ModI per consentire di specificare le seguenti informazioni (Fig. 3.52):

- *Interazione*: specifica il pattern di interazione che si vuole associare alla specifica operation/risorsa
  - *Pattern*: indica il nome del pattern di interazione, a scelta tra Bloccante e Non Bloccante
  - *Tipo*: (solo per il pattern non bloccante) indica se l'interazione prevista è di tipo PUSH (iniziativa del mittente) o PULL (iniziativa del destinatario)
  - *Funzione*: (solo per il pattern non bloccante) indica se l'operation/risorsa ha la funzione di inviare una richiesta, chiedere lo stato di avanzamento dell'elaborazione della risposta o inviare una risposta.
  - *Richiesta Correlata*: (solo per la funzione Richiesta Stato e Risposta) indica l'operation/risorsa correlata che corrisponde all'invio della richiesta.

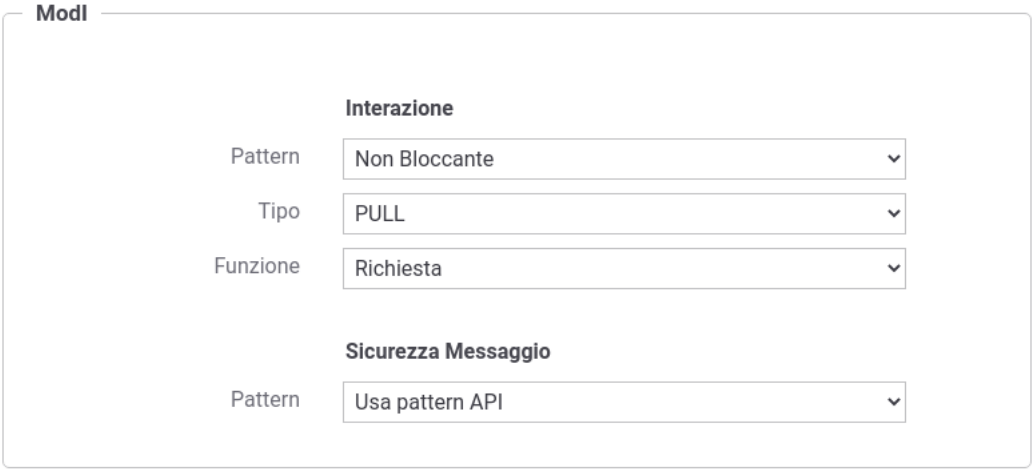


Fig. 3.52: Pattern di interazione ModI per operation/risorse dell'API

Nota: Su API di tipo REST i pattern bloccanti e non bloccanti risultano selezionabili solamente se una risorsa è compatibile con i metodi HTTP e i codici di risposta richiesti dalla specifica.

Nelle sezioni seguenti vengono forniti maggiori dettagli su come siano gestiti i pattern non bloccanti.

3.4.1 Pattern di Interazione PUSH per API SOAP

Il pattern di interazione, denominato PUSH, è utilizzabile nel caso in cui il fruitore abbia a sua volta la possibilità di esporre una interfaccia di servizio per la ricezione delle risposte (Fig. 3.53).

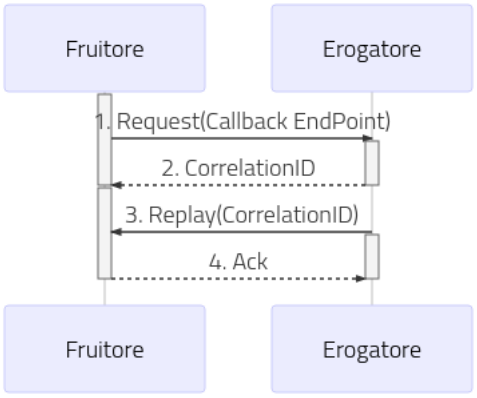


Fig. 3.53: Flusso previsto in un Pattern di Interazione PUSH

Come riportato dalle Linee Guida di Interoperabilità ModI:

- Al passo (1), il fruitore DEVE indicare l'endpoint della callback utilizzando l'header SOAP custom "X-ReplyTo";
- Al passo (2), l'erogatore DEVE fornire insieme all'acknowledgement della richiesta nel body, il correlation ID utilizzando l'header SOAP custom X-Correlation-ID;
- Al passo (3), l'erogatore DEVE riutilizzare lo stesso correlation ID fornito al passo (2) sempre utilizzando l'header SOAP custom X-Correlation-ID;
- Al passo (4), il fruitore DEVE riconoscere tramite un messaggio di acknowledgement il ricevimento della risposta.

Configurazione delle API

Per attuare la configurazione su GovWay si deve procedere con la registrazione delle due API che definiscono il servizio di ricezione della richiesta e il servizio di Callback dove l'erogatore deve inoltrare la risposta.

- Richiesta

Effettuata la registrazione delle API, accedere al dettaglio dell'azione relativa al servizio di richiesta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PUSH" con ruolo "Richiesta" come mostrato nella figura Fig. 3.54:

- Risposta

Successivamente, accedere al dettaglio dell'azione relativa al servizio di callback ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PUSH" con ruolo "Risposta". Definire anche la correlazione verso il servizio e l'azione relativa alla richiesta come mostrato nella figura Fig. 3.55:

Configurazione dell'Erogazione

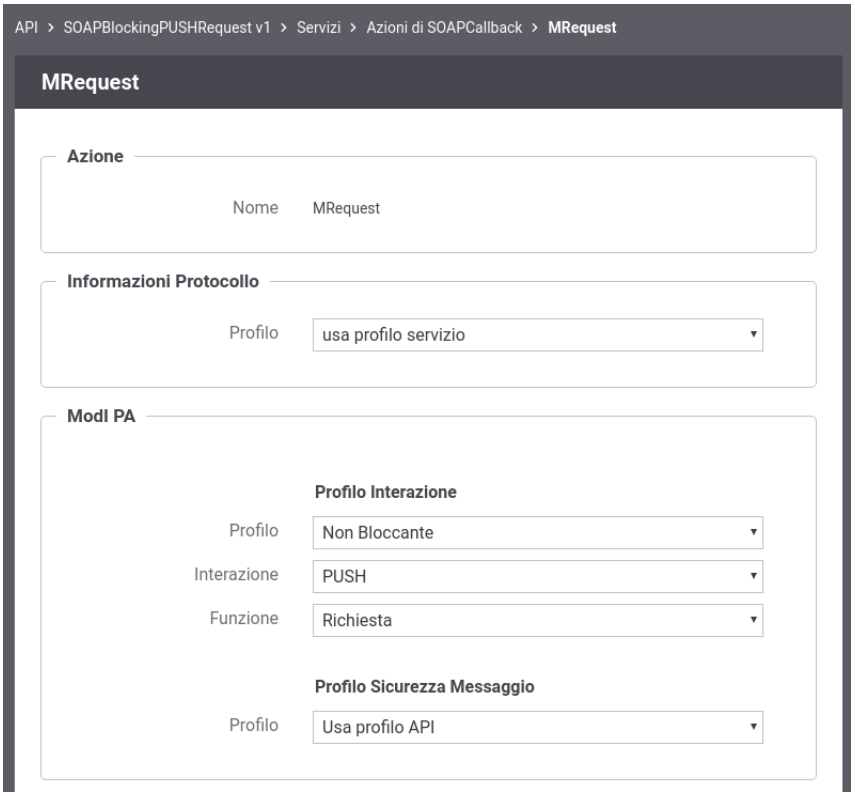


Fig. 3.54: Configurazione della richiesta dell'API SOAP (PUSH)

Sul dominio dell'erogatore deve essere definita sia un'erogazione dell'API relativa al servizio di richiesta che una fruizione del servizio di callback.

- Erogazione del Servizio di Richiesta

Le richieste ricevute sull'erogazione vengono validate da GovWay verificando la presenza dell'header SOAP custom "X-ReplyTo" come previsto dal profilo "ModI". Effettuata la validazione del messaggio di richiesta, eventualmente gestendo anche gli aspetti di sicurezza descritti nella sezione *Sicurezza Messaggio*, GovWay inoltra il messaggio al backend e rimane in attesa dell'acknowledgement.

Ricevuto il messaggio di acknowledgement GovWay verifica la presenza dell'header SOAP custom "X-Correlation-ID". Se tale header non risulta presente viene generato da GovWay impostando come valore l'identificativo della transazione, che è stato inoltrato con la richiesta al backend tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Transaction-ID").

Nota: Header "X-Correlation-ID" generato da GovWay

La generazione dell'header soap "X-Correlation-ID", se non presente, è disabilitabile intervenendo sulla proprietà «org.openspcoop2.protocol.modipa.soap.push.request.correlationId.header.useTransactionIdIfNotExists» presente nel file "/etc/govway/modipa_local.properties" (si assume che "/etc/govway" sia la directory di configurazione indicata in fase di installazione). Se si disabilita la proprietà, GovWay termina con errore la transazione se rileva l'assenza dell'header soap "X-Correlation-ID" nel messaggio di acknowledgement ricevuto dal backend.

- Fruizione del Servizio di Callback per la Risposta

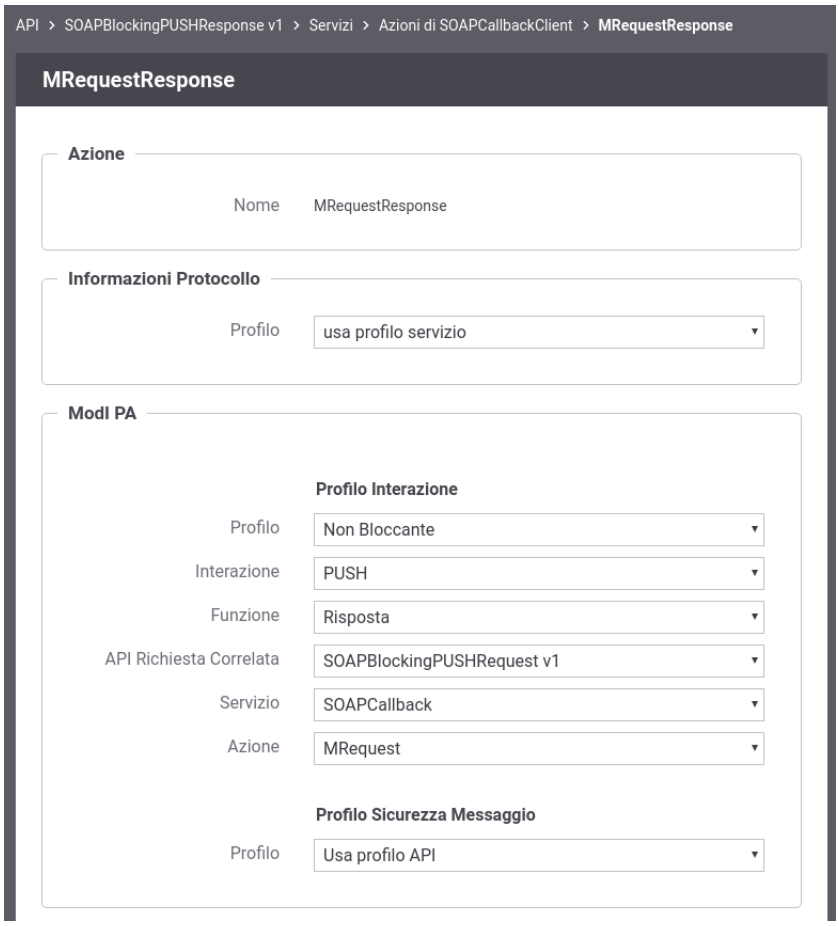


Fig. 3.55: Configurazione della risposta dell'API SOAP (PUSH)

Le risposte devono essere inoltrate dall'applicativo mittente utilizzando la fruizione del servizio di Callback configurata su GovWay. Le risposte vengono validate da GovWay verificando la presenza dell'header SOAP custom "X-Correlation-ID". GovWay permette di fornire l'informazione sull'identificativo di correlazione anche tramite modalità alternative all'header soap (header http, parametri della url...) per poi generare un header soap "X-Correlation-ID" come previsto dalla specifica "ModI" valorizzato con il valore fornito. Le modalità alternative sono le seguenti:

- Header HTTP "X-Correlation-ID"
- Header HTTP "GovWay-Conversation-ID" o parametro della url "govway_conversation_id" previsto per la correlazione tramite identificativo di collaborazione descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di collaborazione nell'API o sulla singola azione come mostrato nella seguente Fig. 3.56:

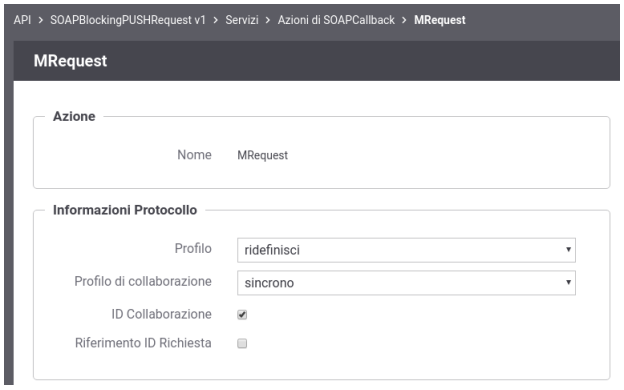


Fig. 3.56: Abilitazione funzionalità di correlazione govway tramite identificativo di colllaborazione

- Header HTTP "GovWay-Relates-To" o parametro della url "govway_relates_to" previsto per la correlazione tramite riferimento della richiesta descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di riferimento alla richiesta nell'API o sulla singola azione come mostrato nella figura Fig. 3.57:

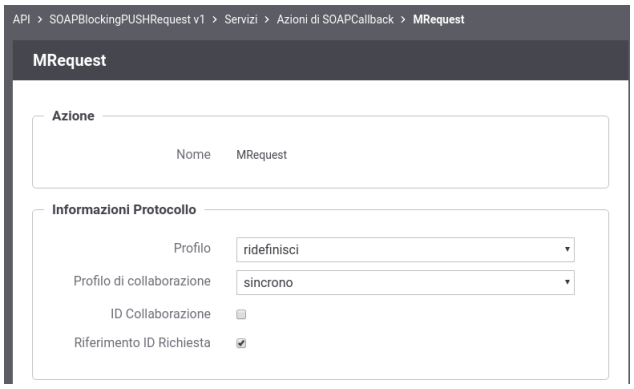


Fig. 3.57: Abilitazione funzionalità di correlazione govway tramite identificativo della richiesta

Configurazione della Fruizione

Sul dominio del fruitore deve essere definita sia una fruizione dell'API relativa al servizio di richiesta che un'erogazione del servizio di callback.

- Fruizione del Servizio di Richiesta

Le richieste devono essere inoltrate dall'applicativo mittente utilizzando la fruizione del servizio di richiesta configurata su GovWay. Su ogni richiesta GovWay crea, o ne modifica il valore se già presente, dell'header soap "X-ReplyTo" previsto dal profilo "ModI". L'header viene valorizzato con l'url di invocazione utilizzabile dalla controparte per invocare il servizio di callback configurato su GovWay.

Nota: Header "X-ReplyTo" generato da GovWay

La valorizzazione dell'header soap "X-ReplyTo" da parte di GovWay è disabilitabile intervenendo sulla proprietà «org.openspcoop2.protocol.modipa.soap.push.replyTo.header.updateOrCreate» presente nel file "/etc/govway/modipa_local.properties" (si assume che "/etc/govway" sia la directory di configurazione indicata in fase di installazione). Se si disabilita la proprietà, GovWay termina con errore la transazione se rileva l'assenza dell'header soap "X-ReplyTo" nel messaggio di richiesta ricevuto dal backend.

Il messaggio di acknowledgement ricevuto viene anch'esso validato al fine di verificare la presenza dell'header soap "X-Correlation-ID" come previsto dalla specifica "ModI". L'informazione sull'id di correlazione è ottenibile dall'applicativo mittente sulla risposta, oltre che tramite l'header soap "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

- Erogazione del Servizio di Callback per la Risposta

Le risposte ricevute sull'erogazione del servizio di Callback vengono validate da GovWay verificando la presenza dell'header SOAP custom "X-Correlation-ID" come previsto dal profilo "ModI". Effettuata la validazione del messaggio di risposta, eventualmente gestendo anche gli aspetti di sicurezza descritti nella sezione *Sicurezza Messaggio*, GovWay inoltra il messaggio al backend e rimane in attesa dell'acknowledgement. L'informazione sull'id di correlazione è inoltrato al backend, oltre che tramite l'header soap "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

3.4.2 Pattern di Interazione PUSH per API REST

Il pattern di interazione, denominato PUSH, è utilizzabile nel caso in cui il fruitore abbia a sua volta la possibilità di esporre una interfaccia di servizio per la ricezione delle risposte (Fig. 3.58).

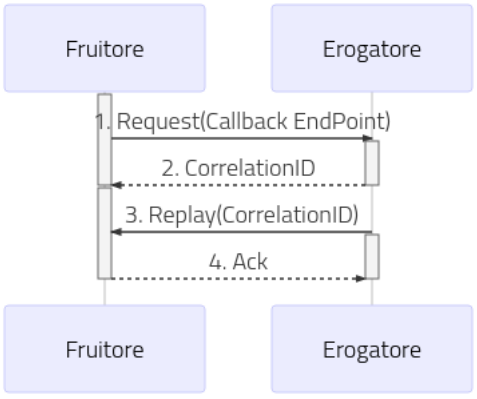


Fig. 3.58: Flusso previsto in un Pattern di Interazione PUSH

Come riportato dalle Linee Guida di Interoperabilità ModI:

- Al passo (1), il fruitore DEVE indicare l'endpoint della callback utilizzando l'header HTTP custom X-ReplyTo ed usando HTTP method POST;
- Al passo (2), l'erogatore DEVE fornire insieme all'acknowledgement della richiesta nel body, il correlation ID utilizzando l'header HTTP custom X-Correlation-ID; Il codice HTTP di stato DEVE essere HTTP status 202 Accepted a meno che non si verifichino errori;
- Al passo (3), l'erogatore DEVE riutilizzare lo stesso correlation ID fornito al passo (2) sempre utilizzando l'header HTTP custom X-Correlation-ID; Il verbo HTTP utilizzato deve essere POST;
- Al passo (4), il fruitore DEVE riconoscere tramite un messaggio di acknowledgement il ricevimento della risposta; Il codice HTTP di stato DEVE essere HTTP status 200 OK a meno che non si verifichino errori.

Configurazione delle API

Per attuare la configurazione su GovWay si deve procedere con la registrazione delle due API che definiscono il servizio di ricezione della richiesta e il servizio di Callback dove l'erogatore deve inoltrare la risposta.

- Richiesta

Effettuata la registrazione delle API, accedere al dettaglio della risorsa relativa al servizio di richiesta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PUSH" con ruolo "Richiesta" come mostrato nella figura Fig. 3.59:

- Risposta

Successivamente, accedere al dettaglio della risorsa relativa al servizio di callback ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PUSH" con ruolo "Risposta". Definire anche la correlazione verso l'API e l'azione relativa alla richiesta come mostrato nella figura Fig. 3.60:

Configurazione dell'Erogazione

Sul dominio dell'erogatore deve essere definita sia un'erogazione dell'API relativa al servizio di richiesta che una fruizione del servizio di callback.

- Erogazione del Servizio di Richiesta

Le richieste ricevute sull'erogazione vengono validate da GovWay verificando la presenza dell'header HTTP custom "X-ReplyTo" come previsto dal profilo "ModI". Effettuata la validazione del messaggio di richiesta, eventualmente gestendo anche gli aspetti di sicurezza descritti nella sezione *Sicurezza Messaggio*, GovWay inoltra il messaggio al backend e rimane in attesa dell'acknowledgement.

Ricevuto il messaggio di acknowledgement GovWay verifica che il codice HTTP di stato sia 202 e verifica la presenza dell'header HTTP custom "X-Correlation-ID". Se tale header non risulta presente viene generato da GovWay impostando come valore l'identificativo della transazione, che è stato inoltrato con la richiesta al backend tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Transaction-ID").

Nota: Header "X-Correlation-ID" generato da GovWay

La generazione dell'header HTTP "X-Correlation-ID", se non presente, è disabilitabile intervenendo sulla proprietà «org.openspcoop2.protocol.modipa.rest.push.request.correlationId.header.useTransactionIdIfNotExists» presente nel file "/etc/govway/modipa_local.properties" (si assume che "/etc/govway" sia la directory di configurazione indicata in fase di installazione). Se si disabilita la proprietà, GovWay termina con errore la transazione se rileva l'assenza dell'header HTTP "X-Correlation-ID" nel messaggio di acknowledgement ricevuto dal backend.

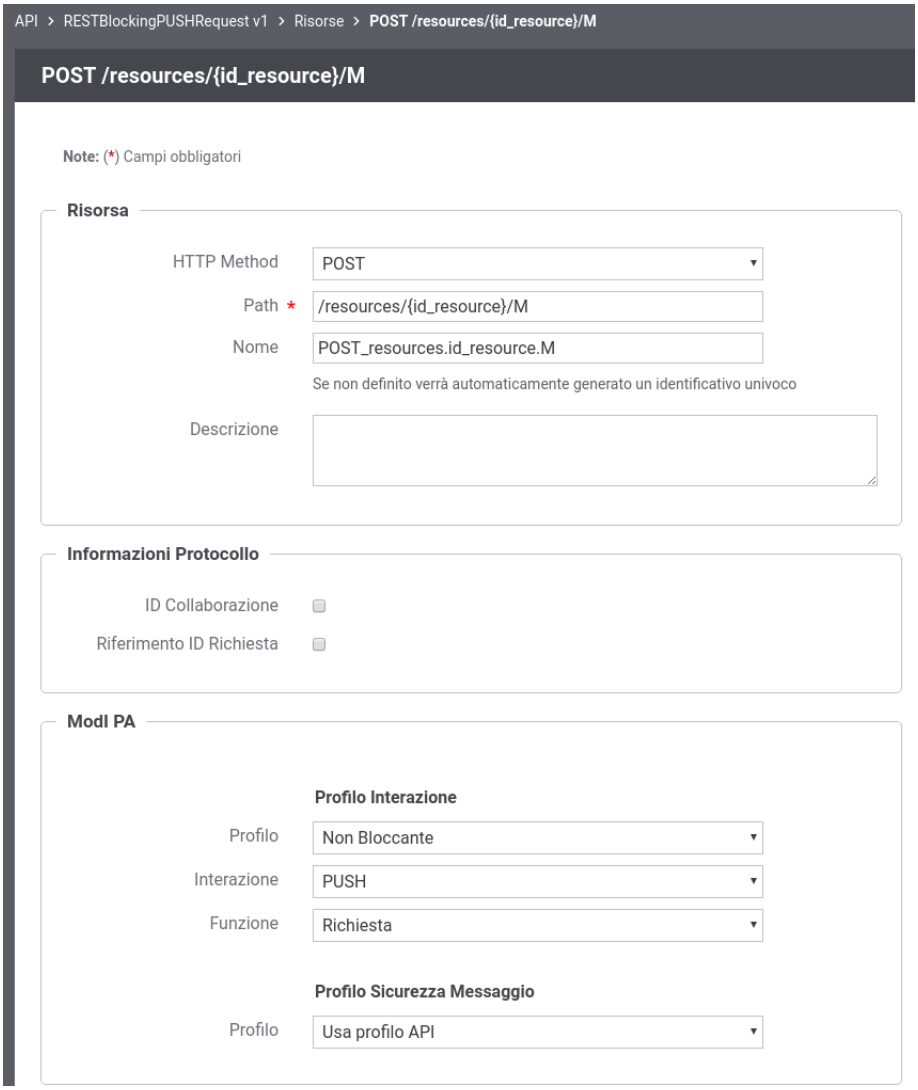


Fig. 3.59: Configurazione della richiesta dell'API REST (PUSH)

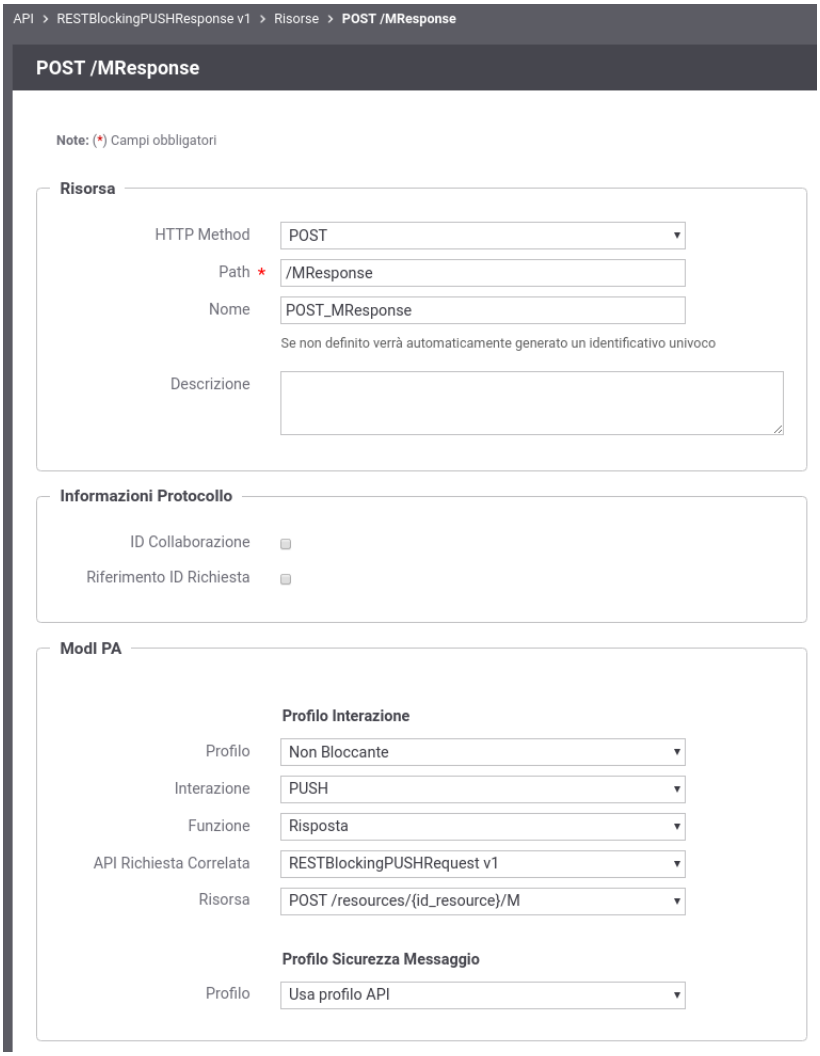


Fig. 3.60: Configurazione della risposta dell'API REST (PUSH)

- Fruizione del Servizio di Callback per la Risposta

Le risposte devono essere inoltrate dall'applicativo mittente utilizzando la fruizione del servizio di Callback configurata su GovWay. Le risposte vengono validate da GovWay verificando la presenza dell'header HTTP custom "X-Correlation-ID". GovWay permette di fornire l'informazione sull'identificativo di correlazione anche tramite modalità alternative all'header HTTP custom per poi generarlo come previsto dalla specifica "ModI" valorizzato con il valore fornito. Le modalità alternative sono le seguenti:

- Header HTTP "GovWay-Conversation-ID" o parametro della url "govway_conversation_id" previsto per la correlazione tramite identificativo di collaborazione descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di collaborazione nell'API o sulla singola azione come mostrato nella figura Fig. 3.61:

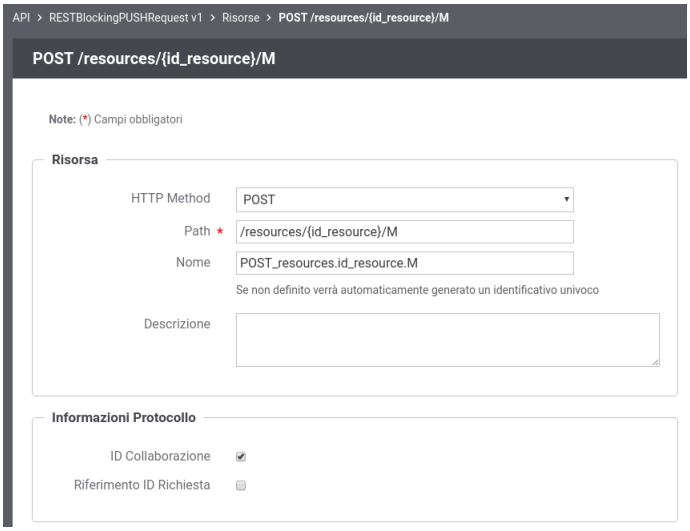


Fig. 3.61: Abilitazione funzionalità di correlazione govway tramite identificativo di colllaborazione

- Header HTTP "GovWay-Relates-To" o parametro della url "govway_relates_to" previsto per la correlazione tramite riferimento della richiesta descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di riferimento alla richiesta nell'API o sulla singola azione come mostrato nella figura Fig. 3.62:

Configurazione della Fruizione

Sul dominio del fruitore deve essere definita sia una fruizione dell'API relativa al servizio di richiesta che un'erogazione del servizio di callback.

- Fruizione del Servizio di Richiesta

Le richieste devono essere inoltrate dall'applicativo mittente utilizzando la fruizione del servizio di richiesta configurata su GovWay. Su ogni richiesta GovWay crea, o ne modifica il valore se già presente, dell'header HTTP "X-ReplyTo" previsto dal profilo "ModI". L'header viene valorizzato con l'url di invocazione utilizzabile dalla controparte per invocare il servizio di callback configurato su GovWay.

Nota: Header "X-ReplyTo" generato da GovWay

La valorizzazione dell'header HTTP "X-ReplyTo" da parte di GovWay è disabilitabile intervenendo sulla proprietà «org.openspcoop2.protocol.modipa.rest.push.replyTo.header.updateOrCreate» presente nel file "/etc/govway/modipa_local.properties" (si assume che "/etc/govway" sia la directory di configurazione indicata in

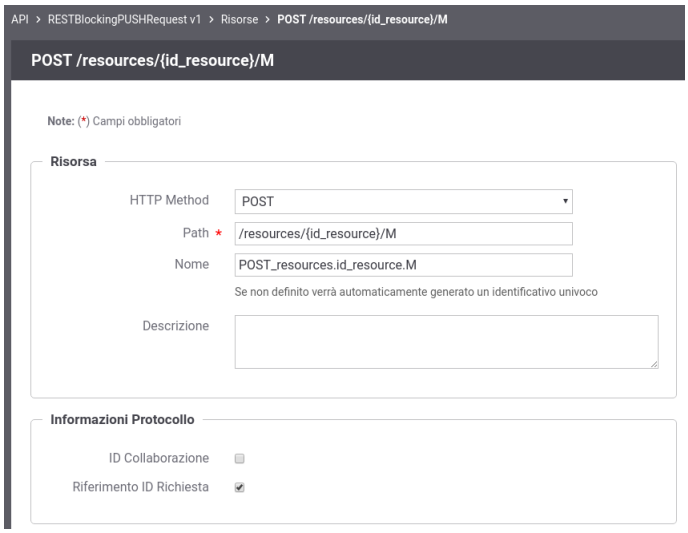


Fig. 3.62: Abilitazione funzionalità di correlazione govway tramite identificativo della richiesta

fase di installazione). Se si disabilita la proprietà, GovWay termina con errore la transazione se rileva l'assenza dell'header HTTP "X-ReplyTo" nel messaggio di richiesta ricevuto dal backend.

Il messaggio di acknowledgement ricevuto viene anch'esso validato al fine di verificare la presenza dell'header HTTP "X-Correlation-ID" come previsto dalla specifica "ModI". L'informazione sull'id di correlazione è ottenibile dall'applicativo mittente sulla risposta, oltre che tramite l'header HTTP "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

- Erogazione del Servizio di Callback per la Risposta

Le risposte ricevute sull'erogazione del servizio di Callback vengono validate da GovWay verificando la presenza dell'header HTTP custom "X-Correlation-ID" come previsto dal profilo "ModI". Effettuata la validazione del messaggio di risposta, eventualmente gestendo anche gli aspetti di sicurezza descritti nella sezione *Sicurezza Messaggio*, GovWay inoltra il messaggio al backend e rimane in attesa dell'acknowledgement. L'informazione sull'id di correlazione è inoltrato al backend, oltre che tramite l'header HTTP "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

3.4.3 Pattern di Interazione PULL per API SOAP

Il pattern di interazione, denominato PULL, prevede che il fruitore non fornisca un indirizzo di callback, mentre l'erogatore fornisce un indirizzo interrogabile per verificare lo stato di processamento di una richiesta e, al fine dell'elaborazione della stessa, il risultato (Fig. 3.63).

Come riportato dalle Linee Guida di Interoperabilità ModI:

- L'interfaccia di servizio dell'erogatore fornisce tre metodi differenti al fine di inoltrare una richiesta, controllarne lo stato ed ottenerne il risultato
- Al passo (1), il fruitore effettua una richiesta;
- Al passo (2), l'erogatore DEVE fornire insieme all'acknowledgement della richiesta nel body, un correlation ID riportato nel header custom SOAP X-Correlation-ID;

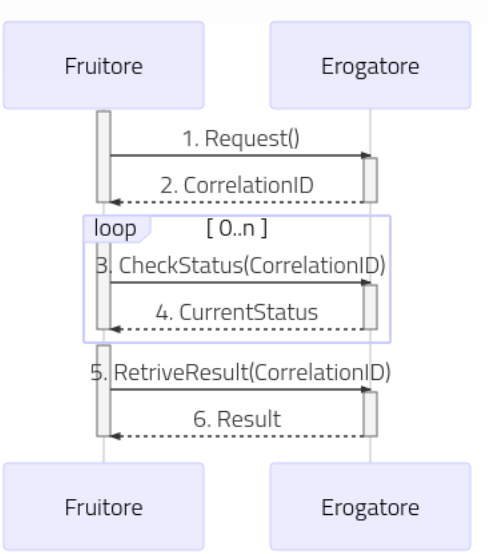


Fig. 3.63: Flusso previsto in un Pattern di Interazione PULL per API SOAP

- Al passo (3), il fruitore DEVE utilizzare i l correlation ID ottenuto al passo (2) per richiedere lo stato di processamento di una specifica richiesta;
- Al passo (4) l'erogatore, quando il processamento non si è ancora concluso fornisce informazioni circa lo stato della lavorazione della richiesta, quando invece il processamento si è concluso risponde indicando in maniera esplicita il completamento;
- Al passo (5), il fruitore utilizza il correlation ID di cui al passo (2) al fine di richiedere il risultato della richiesta;
- Al passo (6), l'erogatore fornisce il risultato del processamento.

Configurazione delle API

Per attuare la configurazione su GovWay si deve procedere con la registrazione dell'API che deve contenere i tre metodi differenti descritti precedentemente.

- Richiesta

Effettuata la registrazione delle API, accedere al dettaglio dell'azione corrispondente alla richiesta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Richiesta" come mostrato nella figura Fig. 3.64:

- Richiesta Stato

Successivamente, accedere al dettaglio dell'azione che consente di richiedere lo stato di processamento ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Richiesta Stato". Definire anche la correlazione verso l'azione relativa alla richiesta come mostrato nella figura Fig. 3.65:

- Risposta

Accedere al dettaglio dell'azione corrispondente alla risposta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Risposta". Definire anche la correlazione verso l'azione relativa alla richiesta come mostrato nella figura Fig. 3.66:

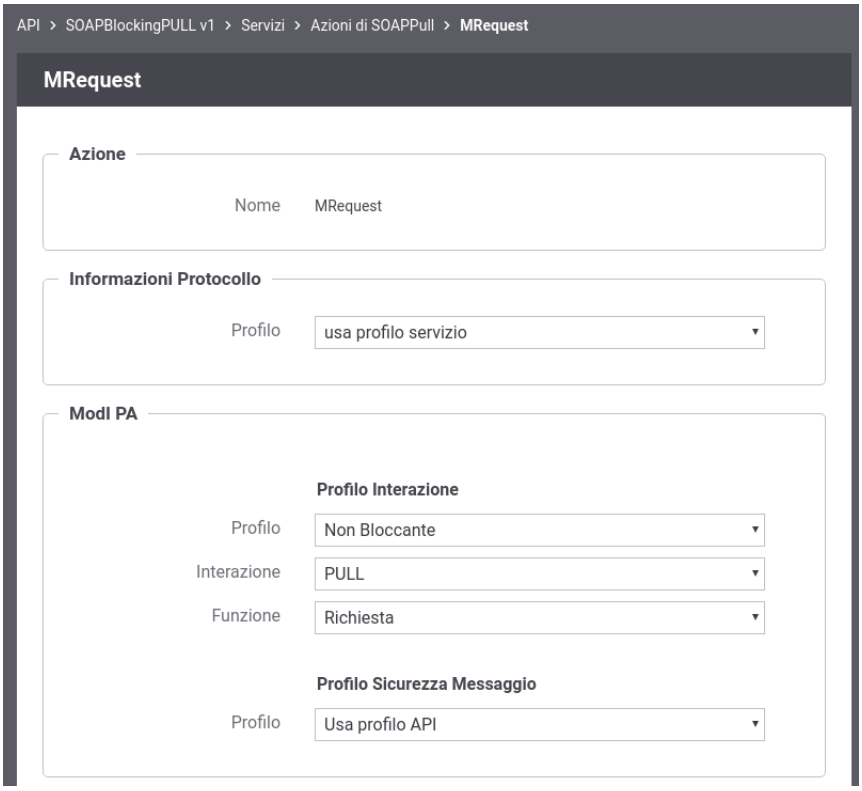


Fig. 3.64: Configurazione della richiesta dell'API SOAP (PULL)

Configurazione dell'Erogazione

Sul dominio dell'erogatore deve essere definita l'erogazione dell'API.

- Richiesta

Le richieste ricevute sull'erogazione vengono inoltrate al backend da GovWay rimanendo poi in attesa dell'acknowledgement.

Ricevuto il messaggio di acknowledgement GovWay verifica la presenza dell'header SOAP custom "X-Correlation-ID". Se tale header non risulta presente viene generato da GovWay impostando come valore l'identificativo della transazione, che è stato inoltrato con la richiesta al backend tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Transaction-ID").

Nota: Header "X-Correlation-ID" generato da GovWay

La generazione dell'header soap "X-Correlation-ID", se non presente, è disabilitabile intervenendo sulla proprietà «org.openspcoop2.protocol.modipa.soap.pull.request.correlationId.header.useTransactionIdIfNotExists» presente nel file "/etc/govway/modipa_local.properties" (si assume che "/etc/govway" sia la directory di configurazione indicata in fase di installazione). Se si disabilita la proprietà, GovWay termina con errore la transazione se rileva l'assenza dell'header soap "X-Correlation-ID" nel messaggio di acknowledgement ricevuto dal backend.

- Richiesta Stato di Processamento

Le richieste che richiedono uno stato del processamento vengono validate da GovWay verificando la presenza dell'-header soap "X-Correlation-ID" come previsto dal profilo "ModI". Effettuata la validazione del messaggio di richiesta, eventualmente gestendo anche gli aspetti di sicurezza descritti nella sezione *Sicurezza Messaggio*, GovWay inoltra il

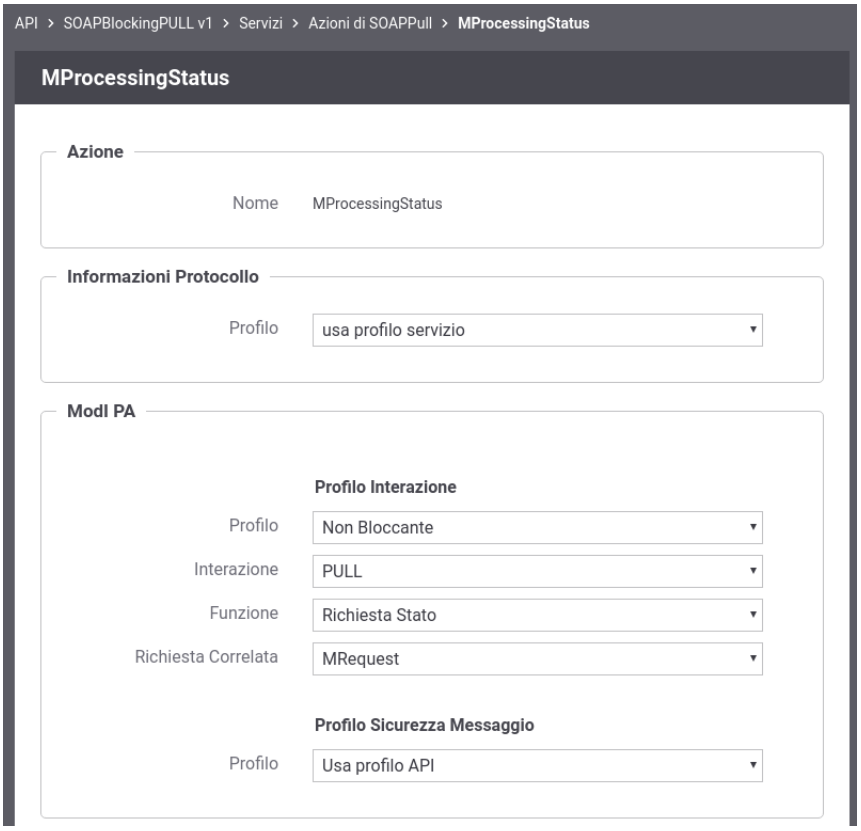


Fig. 3.65: Configurazione della richiesta stato di processamento dell'API SOAP (PULL)

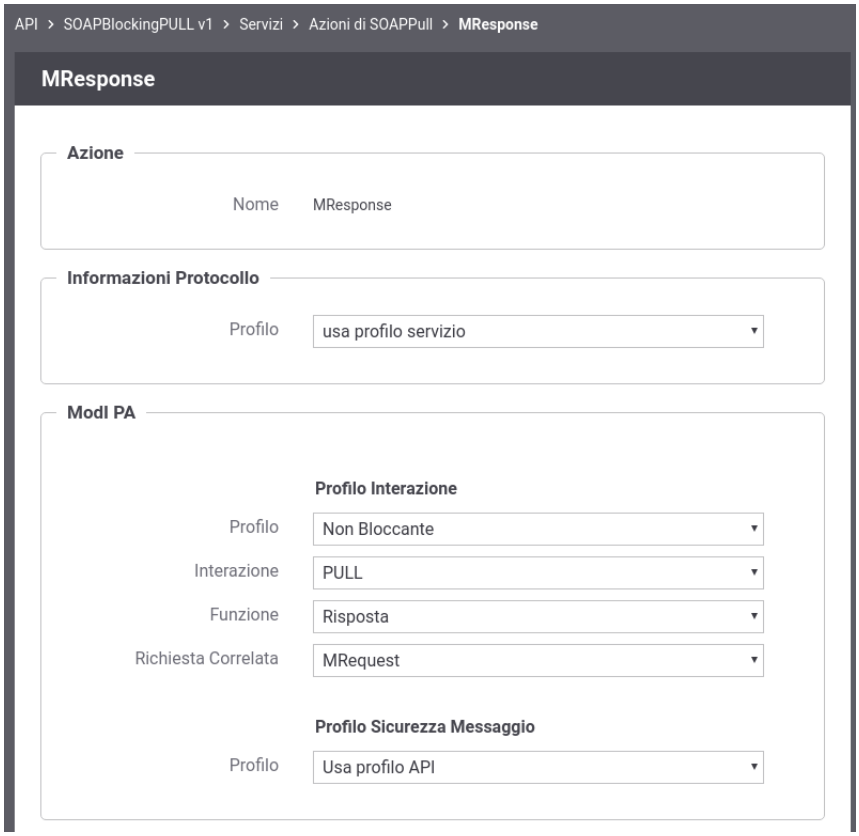


Fig. 3.66: Configurazione della risposta dell'API SOAP (PUSH)

messaggio al backend e rimane in attesa dell'acknowledgement. L'informazione sull'id di correlazione è inoltrato al backend, oltre che tramite l'header soap "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

- Risposta

Le risposte vengono gestite da GovWay in maniera simile a quanto indicato per le richieste di stato del processamento.

Configurazione della Fruizione

Sul dominio del fruitore deve essere definita una fruizione dell'API.

- Richiesta

Le richieste devono essere inoltrate dall'applicativo mittente utilizzando la fruizione dell'API configurata su GovWay.

Il messaggio di acknowledgement ricevuto viene validato al fine di verificare la presenza dell'header soap "X-Correlation-ID" come previsto dalla specifica "ModI". L'informazione sull'id di correlazione è ottenibile dall'applicativo mittente sulla risposta, oltre che tramite l'header soap "X-Correlation-ID", anche tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

- Richiesta Stato di Processamento

Le richieste che richiedono uno stato del processamento devono essere inoltrate dall'applicativo mittente utilizzando la fruizione dell'API configurata su GovWay. Le richieste vengono validate da GovWay verificando la presenza dell'header soap "X-Correlation-ID". GovWay permette di fornire l'informazione sull'identificativo di correlazione anche tramite modalità alternative all'header soap per poi generarlo come previsto dalla specifica "ModI" valorizzato con il valore fornito. Le modalità alternative sono le seguenti:

- Header HTTP "X-Correlation-ID"
- Header HTTP "GovWay-Conversation-ID" o parametro della url "govway_conversation_id" previsto per la correlazione tramite identificativo di collaborazione descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di collaborazione nell'API o sulla singola azione come mostrato nella figura Fig. 3.67:

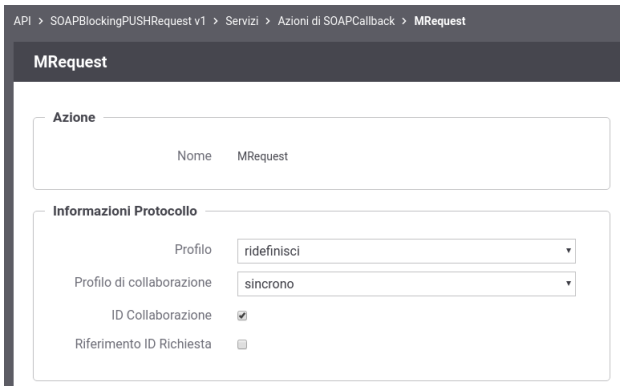


Fig. 3.67: Abilitazione funzionalità di correlazione govway tramite identificativo di colllaborazione

- Header HTTP "GovWay-Relates-To" o parametro della url "govway_relates_to" previsto per la correlazione tramite riferimento della richiesta descritta nella sezione *Correlazione tra transazioni differenti*. Questa modalità richiede che sia abilitata l'indicazione dell'identificativo di riferimento alla richiesta nell'API o sulla singola azione come mostrato nella figura Fig. 3.68:

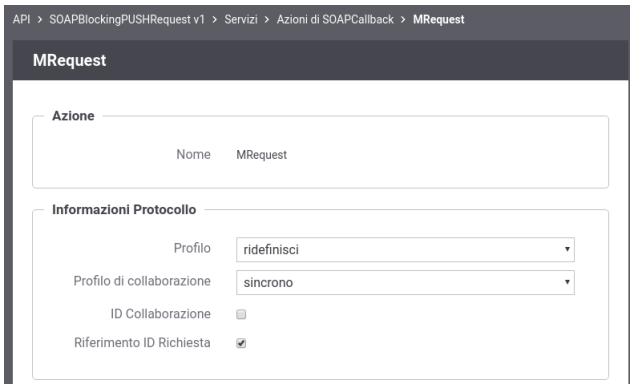


Fig. 3.68: Abilitazione funzionalità di correlazione govway tramite identificativo della richiesta

- Risposta

Le risposte vengono gestite da GovWay in maniera simile a quanto indicato per le richieste di stato del processamento.

3.4.4 Pattern di Interazione PULL per API REST

Il pattern di interazione, denominato PULL, prevede che il fruitore non fornisca un indirizzo di callback, mentre l'erogatore fornisce un indirizzo interrogabile per verificare lo stato di processamento di una richiesta e, al fine dell'elaborazione della stessa, il risultato (Fig. 3.69).

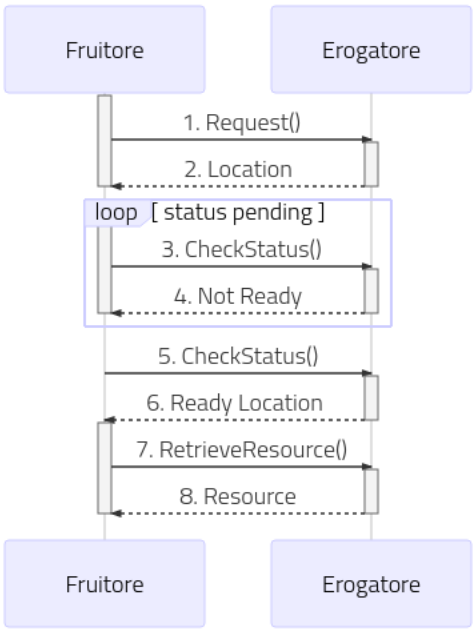


Fig. 3.69: Flusso previsto in un Pattern di Interazione PULL per API REST

Come riportato dalle Linee Guida di Interoperabilità ModI:

- L'interfaccia di servizio dell'erogatore fornisce tre metodi differenti al fine di inoltrare una richiesta, controllarne lo stato ed ottenerne il risultato

- Al passo (1), il fruitore DEVE utilizzare il verbo HTTP POST;
- Al passo (2), l'erogatore DEVE fornire insieme all'acknowledgement della richiesta, un percorso di risorsa per interrogare lo stato di processamento utilizzando HTTP header Location ; Il codice HTTP di stato DEVE essere HTTP status 202 Accepted a meno che non si verifichino errori;
- Al passo (3), il fruitore DEVE utilizzare il percorso di cui al passo (2) per richiedere lo stato della risorsa; Il verbo HTTP utilizzato deve essere GET;
- Al passo (4) l'erogatore indica che la risorsa non è ancora pronta, fornendo informazioni circa lo stato della lavorazione della richiesta; il codice HTTP restituito è HTTP status 200 OK;
- Al passo (6) l'erogatore indica che la risorsa è pronta, utilizzando HTTP header Location ; per indicare il percorso dove recuperare la risorsa, il codice HTTP restituito è HTTP status 303 See Other;
- Al passo (8) l'erogatore risponde con la rappresentazione della risorsa,Il codice HTTP restituito è HTTP status 200 OK;

Configurazione delle API

Per attuare la configurazione su GovWay si deve procedere con la registrazione dell'API che deve contenere le tre risorse differenti descritti precedentemente.

- Richiesta

Effettuata la registrazione delle API, accedere al dettaglio della risorsa corrispondente alla richiesta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Richiesta" come mostrato nella figura Fig. 3.70:

- Richiesta Stato

Successivamente, accedere al dettaglio dell'azione che consente di richiedere lo stato di processamento ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Richiesta Stato". Definire anche la correlazione verso la risorsa relativa alla richiesta come mostrato nella figura Fig. 3.71:

- Risposta

Accedere al dettaglio dell'azione corrispondente alla risposta ed impostare nella sezione "ModI" un pattern di interazione non bloccante "PULL" con ruolo "Risposta". Definire anche la correlazione verso la risorsa relativa alla richiesta come mostrato nella figura Fig. 3.72:

Configurazione dell'Erogazione

Sul dominio dell'erogatore deve essere definita l'erogazione dell'API.

- Richiesta

Le richieste ricevute sull'erogazione vengono inoltrate al backend da GovWay rimanendo poi in attesa dell'acknowledgement.

Ricevuto il messaggio di acknowledgement GovWay verifica che il codice HTTP di stato sia 202 e verifica la presenza dell'header HTTP "Location".

- Richiesta Stato di Processamento

Le richieste che richiedono uno stato del processamento vengono validate da GovWay verificando che il codice HTTP di stato sia 200 (risposta non ancora pronta) o 303 (risposta pronta ad essere recuperata). Nel caso il codice HTTP sia 303 viene anche verificata la presenza dell'header HTTP "Location".

- Risposta

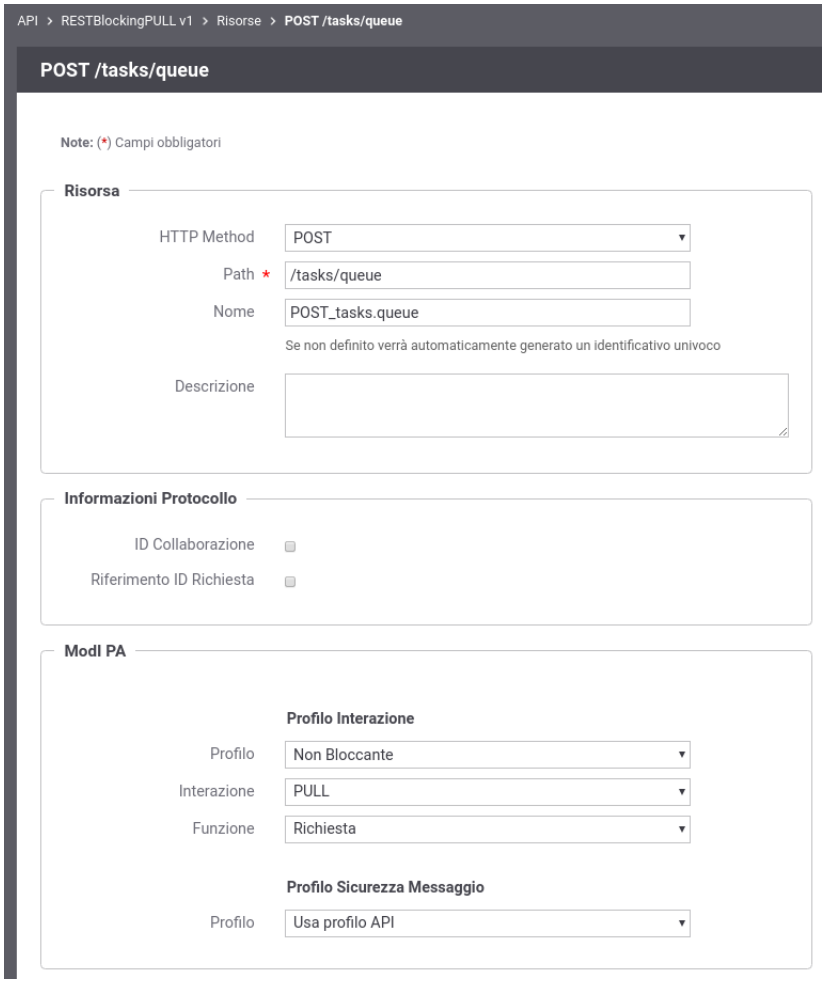


Fig. 3.70: Configurazione della richiesta dell'API REST (PULL)

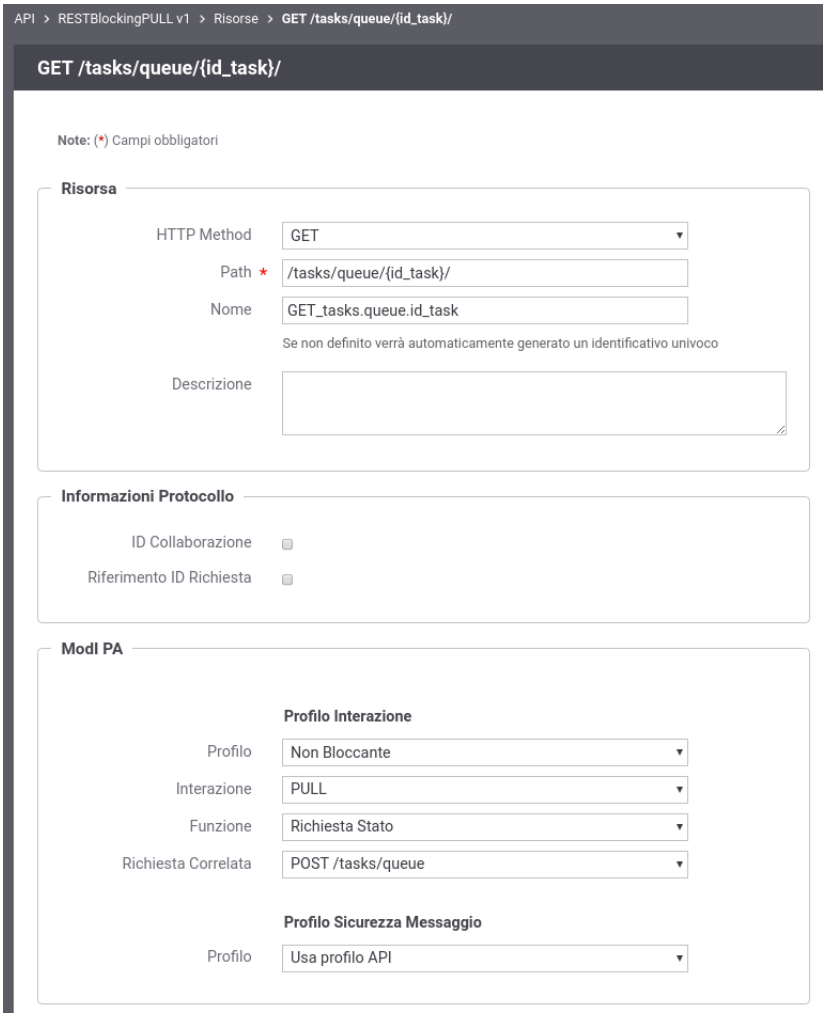


Fig. 3.71: Configurazione della richiesta stato di processamento dell'API REST (PULL)

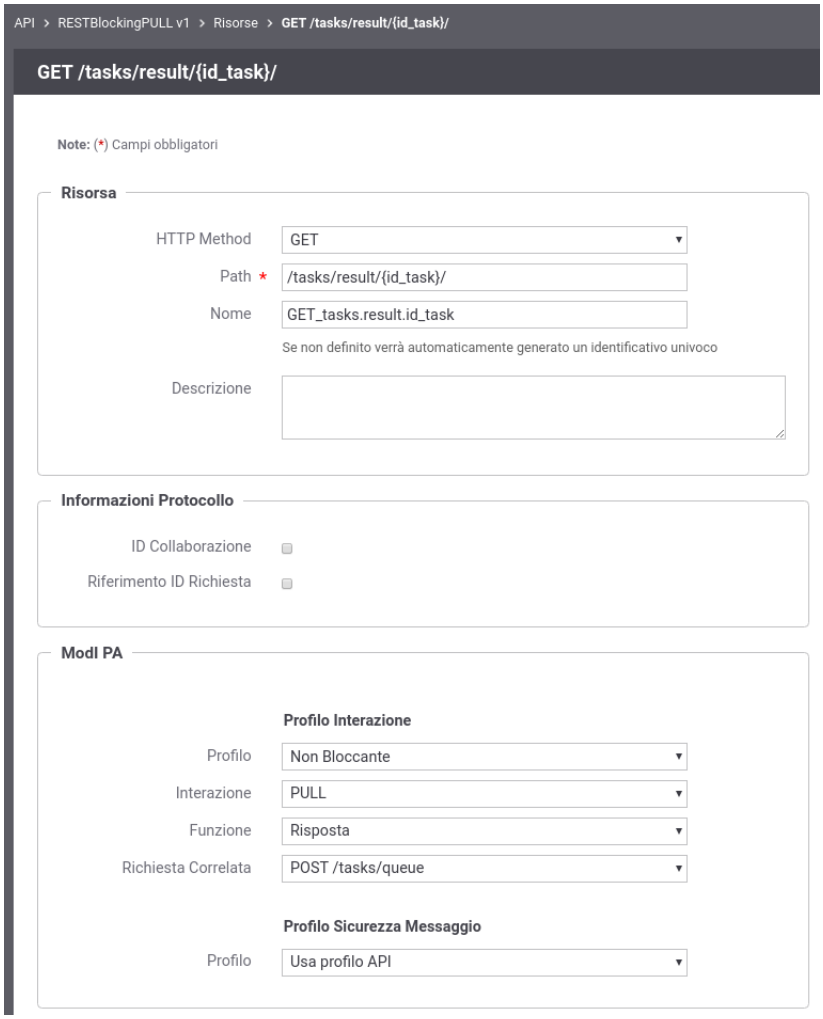


Fig. 3.72: Configurazione della risposta dell'API REST (PUSH)

GovWay valida le risposte verificando che il codice HTTP di stato sia 200.

Nota: Id Correlazione

GovWay estrae dal valore presente nell'header "Location" (per la richiesta e la richiesta stato) e dall'endpoint (per la risposta) l'identificativo di correlazione al fine di correlare la richiesta con le successive operazioni.

Configurazione della Fruizione

Sul dominio del fruitore deve essere definita una fruizione dell'API.

- Richiesta

Le richieste devono essere inoltrate dall'applicativo mittente utilizzando la fruizione dell'API configurata su GovWay.

Il messaggio di acknowledgement ricevuto viene validato al fine di verificare la presenza dell'header http "Location" come previsto dalla specifica "ModI". L'informazione sull'id di correlazione è ottenibile dall'applicativo mittente sulla risposta tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta inoltrata dal gateway al server* e *Altri header di Integrazione* (per default tramite l'header http "GovWay-Conversation-ID").

- Richiesta Stato di Processamento e Risposta

Le successive operazioni devono essere inoltrate dall'applicativo mittente utilizzando la fruizione dell'API configurata su GovWay.

Nota: Id Correlazione

GovWay estrae dal valore presente nell'header "Location" (per la richiesta) e dall'endpoint (per la richiesta stato e per la risposta) l'identificativo di correlazione al fine di correlare la richiesta con le successive operazioni.

CAPITOLO 4

Profilo “eDelivery”

Il profilo eDelivery consente di produrre configurazioni di scenari di interoperabilità che si basano sullo standard europeo eDelivery. Per rendere il trattamento dei messaggi conforme a tale standard, GovWay si interfaccia ad una installazione del software Domibus (https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/Domibus).

Il processo di configurazione rimane strutturalmente analogo a quanto già descritto per la modalità API Gateway. Sono però presenti proprietà specifiche del contesto eDelivery i cui valori devono essere forniti affinché il dialogo con l’access point Domibus possa essere realizzato correttamente.

Nel seguito andiamo a descrivere i passi di configurazione evidenziando, per differenza con il caso API Gateway, gli elementi di eDelivery che dovranno essere gestiti. Al termine della configurazione è necessario procedere con l’export dei dati in formato *PMODE*. Il file prodotto è quello necessario per permettere la configurazione dell’access point Domibus.

4.1 Passi preliminari di configurazione

Per gestire in maniera più semplice i passi di configurazione dei servizi eDelivery è consigliabile impostare l’opportuna modalità operativa della govwayConsole selezionando la voce *eDelivery* sul selettore di modalità presente nella testata dell’applicazione.

Prima di procedere con la configurazione dei servizi si devono verificare i dati relativi ai soggetti interlocutori. Nel caso del soggetto interno al proprio dominio, i dati di configurazione possono essere gestiti alla sezione *Configurazione > Generale* (Fig. 4.1).

Sono presenti valori iniziali, inseriti dal processo di installazione, che devono essere verificati ed eventualmente aggiornati:

- *Base URL Erogazione*: Indirizzo pubblico del Domibus per la ricezione dei messaggi sul canale eDelivery.
- *Base URL Fruizione*: Indirizzo del servizio di GovWay riservato ai client per l’invio di messaggi sul canale eDelivery.

Tramite il collegamento *Visualizza Dati Soggetto* è possibile accedere alla conffigurazione del soggetto interno (Fig. 4.2).

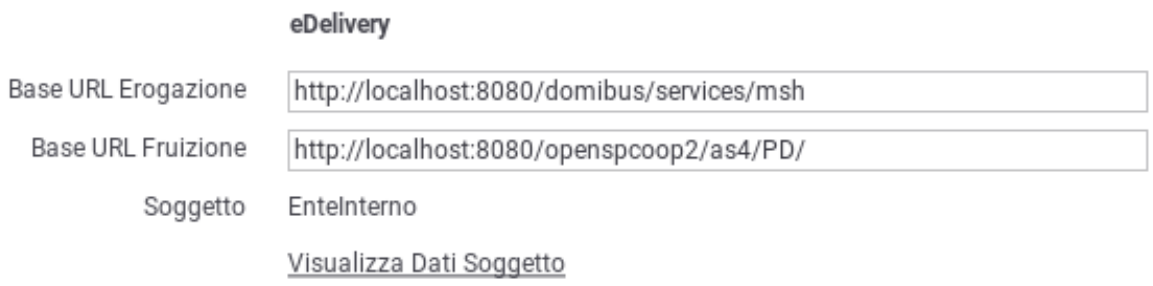


Fig. 4.1: Configurazione delle Base URL eDelivery per il soggetto interno

Le proprietà eDelivery da fornire sono le seguenti:

- *Party Info - Id*: Identificativo del soggetto utilizzato nel canale eDelivery.
- *Party Info - Type Name*: Nome assegnato internamente allo schema indicato nel Type Value.
- *Party Info - Type Value*: Schema di generazione riferito all'identificativo del soggetto eDelivery.
- *Party Endpoint - URL*: Indirizzo pubblico del Domibus per la ricezione dei messaggi sul canale eDelivery.
- *Party Endpoint - Common Name*: Valore della omonima proprietà del certificato utilizzato dall'access point Domibus cui afferisce. Questo nome coincide con quello dell'access point.

4.2 Erogazione di servizi in modalità eDelivery

Configurare un'erogazione eDelivery permette ad un'applicazione interna di ricevere i messaggi inviati da un generico access point eDelivery esterno.

Il primo passo di configurazione prevede che venga censito il soggetto esterno mittente dei messaggi. La creazione di tale soggetto si realizza dalla sezione *Registro* > *Soggetti* della govwayConsole, impostando le proprietà eDelivery già descritte nella sezione precedente per il soggetto interno.

Il passo successivo è quello di registrare le API corrispondenti al servizio eDelivery alla sezione *Registro* > *API*. Le proprietà eDelivery, presenti nel form di creazione, sono quelle mostrate in Fig. 4.3.

Le proprietà da specificare sono le seguenti:

- *Service Info - Type*: Identificativo assegnato come tipo del servizio (opzionale).
- *Service Info - Name*: Nome del servizio.
- *Payload Profiles - File*: Campo per l'upload del descrittore XML che rappresenta il formato dei messaggi inviati dal mittente. Campo opzionale, utilizzabile per aggiungere nuovi profili rispetto a quelli già presenti nell'installazione standard di Domibus. Per la specifica del formato XML da adottare si consulti la documentazione ufficiale di Domibus.
- *Properties - File*: Campo per l'upload del descrittore XML che definisce le proprietà custom che saranno presenti nei messaggi inviati dal mittente. Campo opzionale, utilizzabile per aggiungere nuove property rispetto a quelle già presenti nell'installazione standard di Domibus. Per la specifica del formato XML da adottare si consulti la documentazione ufficiale di Domibus.

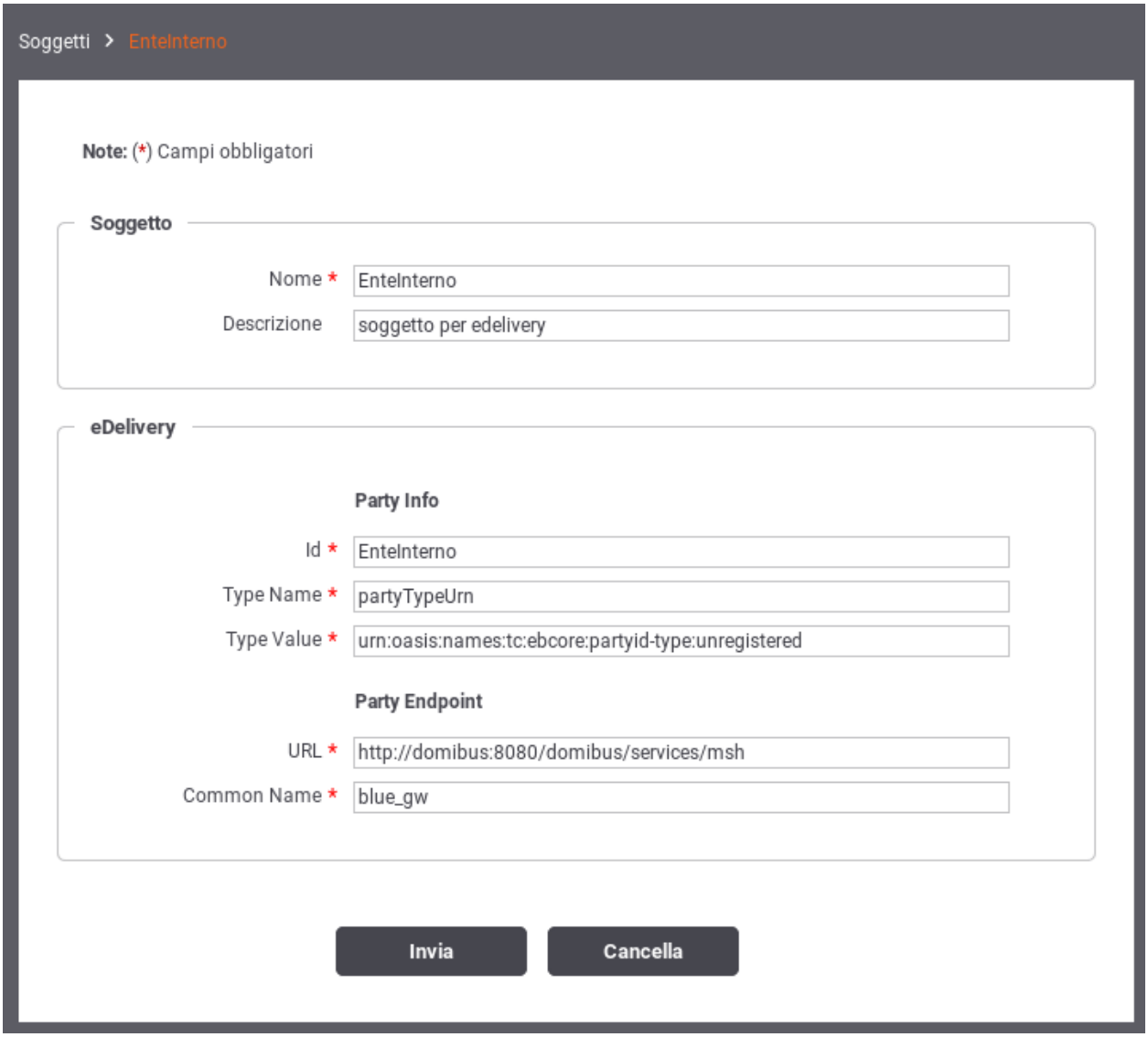


Fig. 4.2: Configurazione delle proprietà eDelivery per il soggetto interno

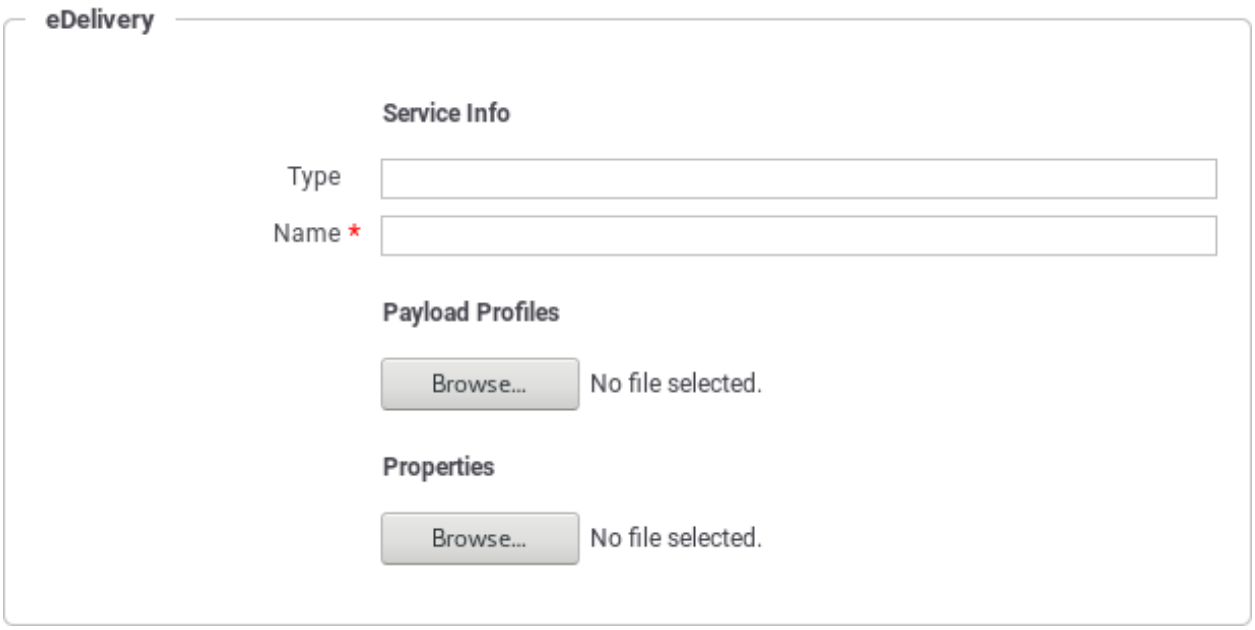


Fig. 4.3: Registrazione API eDelivery - Proprietà specifiche

Dopo aver effettuato il salvataggio è necessario completare la configurazione del servizio utilizzando il link presente nella colonna *Risorse* o *Servizi*, a seconda che si tratti di un servizio Rest o Soap, in corrispondenza dell'elemento presente nell'indice dei servizi. Per ciascuna delle azioni/risorse elencate per il servizio (o create, nel caso che, non disponendo del descrittore del servizio, si proceda con la configurazione manuale delle azioni), si accede al dettaglio per completare la configurazione delle proprerty eDelivery (Fig. 4.4).

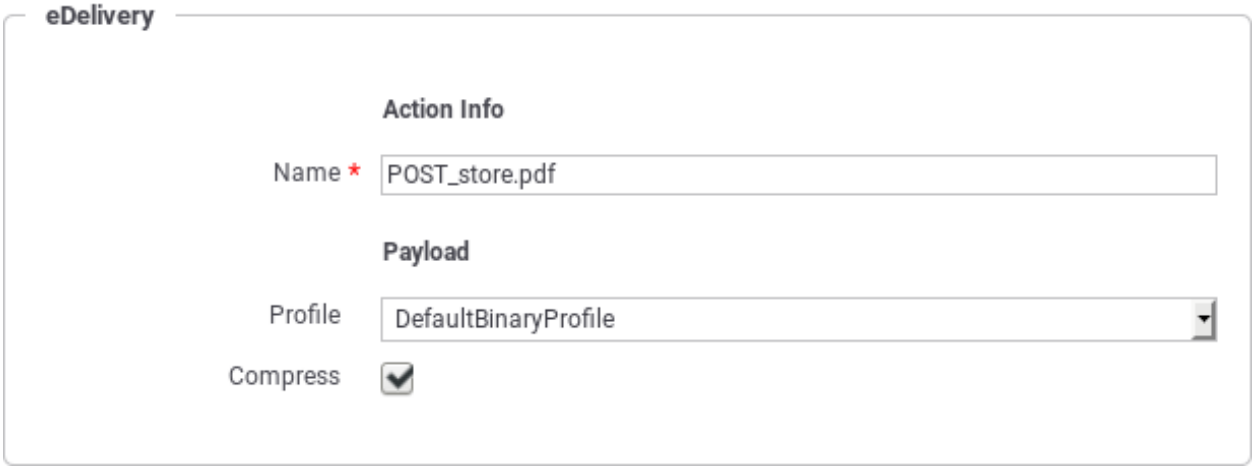


Fig. 4.4: Proprietà eDelivery relative alle azioni delle API

I valori da impostare nel form sono:

- *Action Info* - *Name*: Nome dell'azione.
- *Payload* - *Profile*: Payload Profile, tra quelli disponibili, da utilizzare per l'azione.
- *Payload* - *Compress*: Indicare se l'invio del messaggio farà uso di compressione dei dati.

Dopo aver creato l'API si procede con la configurazione dell'erogazione alla sezione *Registro* > *Erogazioni* della govwayConsole (Fig. 4.5).

Fig. 4.5: Proprietà eDelivery relative all'erogazione del servizio

L'unica impostazione eDelivery da fornire in questo contesto è:

- *Security Profile*: profilo di sicurezza adottato dagli access point durante la comunicazione. E" necessario scegliere tra i valori presenti, che corrispondono alle policy standard, già presenti in Domibus con l'installazione.

Nota: L'endpoint fornito alla voce Connettore sarà quello utilizzato da GovWay per la consegna dei messaggi consegnati all'access point Domibus interno.

Nota: Affinché le configurazioni apportate in modalità eDelivery possano essere attuate sull'access point Domibus è necessario procedere alla generazione del PMODE nel modo descritto alla sezione *Generazione del PMODE Domibus*.

4.3 Fruizione di servizi in modalità eDelivery

Configurare una fruizione eDelivery permette ad un'applicazione interna di inviare messaggi da veicolare verso un generico access point eDelivery esterno.

Il processo di configurazione della fruizione eDelivery prevede inizialmente i medesimi passi già descritti per l'erogazione nella sezione *Erogazione di servizi in modalità eDelivery*. Dovranno quindi essere configurati i dati eDelivery relativi ai soggetti interlocutori, interno ed esterno, dovranno inoltre essere censite le API relative al servizio da fruire.

Dopo aver censito le API si procede con la configurazione della fruizione creando un nuovo elemento nella sezione *Registro* > *Fruizioni* della govwayConsole. Analogamente al caso dell'erogazione si dovrà selezionare la security policy necessaria per gli scambi tra gli access point.

Nota: Affinché le configurazioni apportate in modalità eDelivery possano essere attuate sull'access point Domibus è necessario procedere alla generazione del PMODE nel modo descritto alla sezione *Generazione del PMODE Domibus*.

4.4 Generazione del PMODE Domibus

Affinché il Domibus interno al proprio dominio sia in grado di recepire tutte le configurazioni prodotte nella modalità eDelivery, è necessario che gli venga fornito il relativo file PMODE, così come prevede la configurazione dell'access point eDelivery.

Dopo aver ultimato le configurazioni dei servizi eDelivery, tramite la govwayConsole, si procede all'esportazione del PMODE effettuando i seguenti passaggi (Fig. 4.6):

- Selezionare la voce di menu *Configurazione* > *Esporta*.
- Selezionare la tipologia archivio *domibus-pmode*.
- Premere il pulsante Invia e salvare il file XML che viene restituito.
- Effettuare l'upload del file ottenuto sulla Domibus Console.

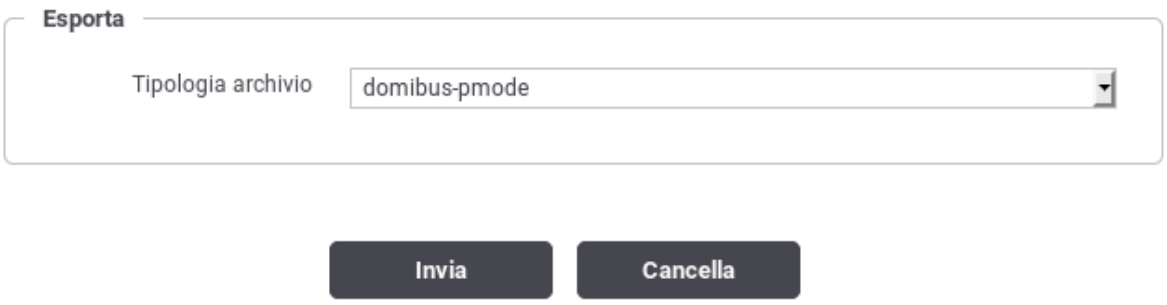


Fig. 4.6: Esportazione del PMODE

CAPITOLO 5

Profilo “SPCoop”

Il profilo SPCoop consente di produrre le configurazioni per i servizi, in accordo alla omonima specifica di cooperazione applicativa della PA italiana. I passi di configurazione, per erogazioni e fruizioni, presentano minime differenze rispetto a quanto descritto per la modalità API Gateway. Nel seguito saranno descritte tali differenze.

5.1 Configurazione di un servizio SPCoop

Il primo passaggio per la configurazione di un servizio SPCoop è quello di creare il relativo Accordo di Servizio. Questi viene creato registrando una nuova API (sezione *Registro* > *API*). Come illustrato nelle figura seguente, la particolarità di questa configurazione, rispetto a quanto descritto in precedenza, risiede nella presenza del campo *Soggetto referente*, nel quale deve essere selezionato uno dei soggetti precedentemente registrati.

Se non viene fornito un WSDL, relativo all’accordo di servizio, è necessario definire manualmente l’interfaccia del servizio, analogamente a quanto descritto in sezione *Configurazione manuale delle interfacce*. In questo caso, l’aggiunta del servizio, comprende i profili di collaborazione asincroni oltre alle caratteristiche aggiuntive specifiche del protocollo SPCoop (vedi sezione *Profili di gestione della busta eGov*). La figura seguente mostra i dettagli di questo caso.

La registrazione di una nuova erogazione o fruizione, presenta le seguenti differenze rispetto a quanto descritto per la modalità API Gateway:

- È presente il campo *Tipo* relativamente al servizio
- È presente il campo *Versione Protocollo* per selezionare la versione della specifica SPCoop adottata.

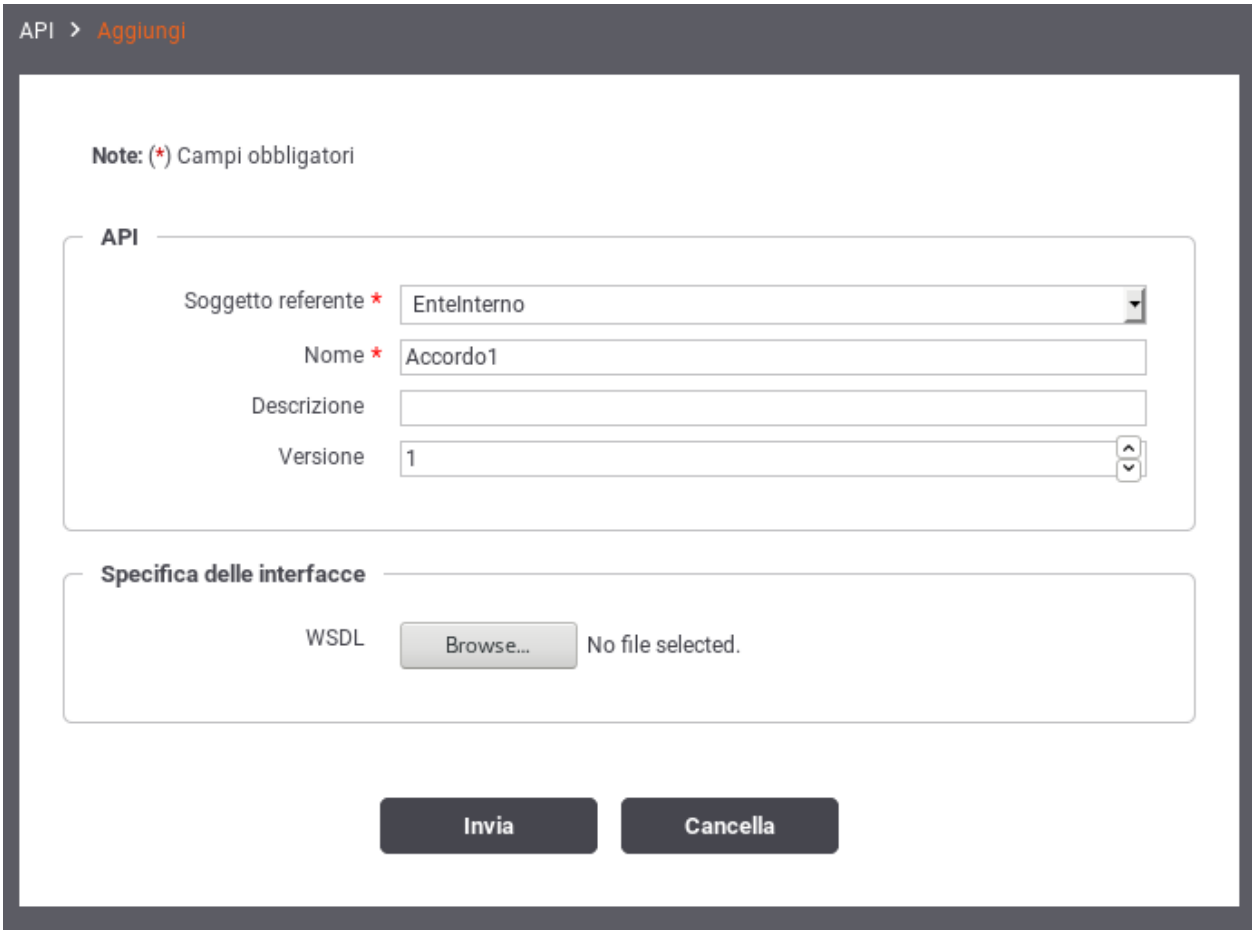


Fig. 5.1: Creazione Accordo di Servizio SPCoop

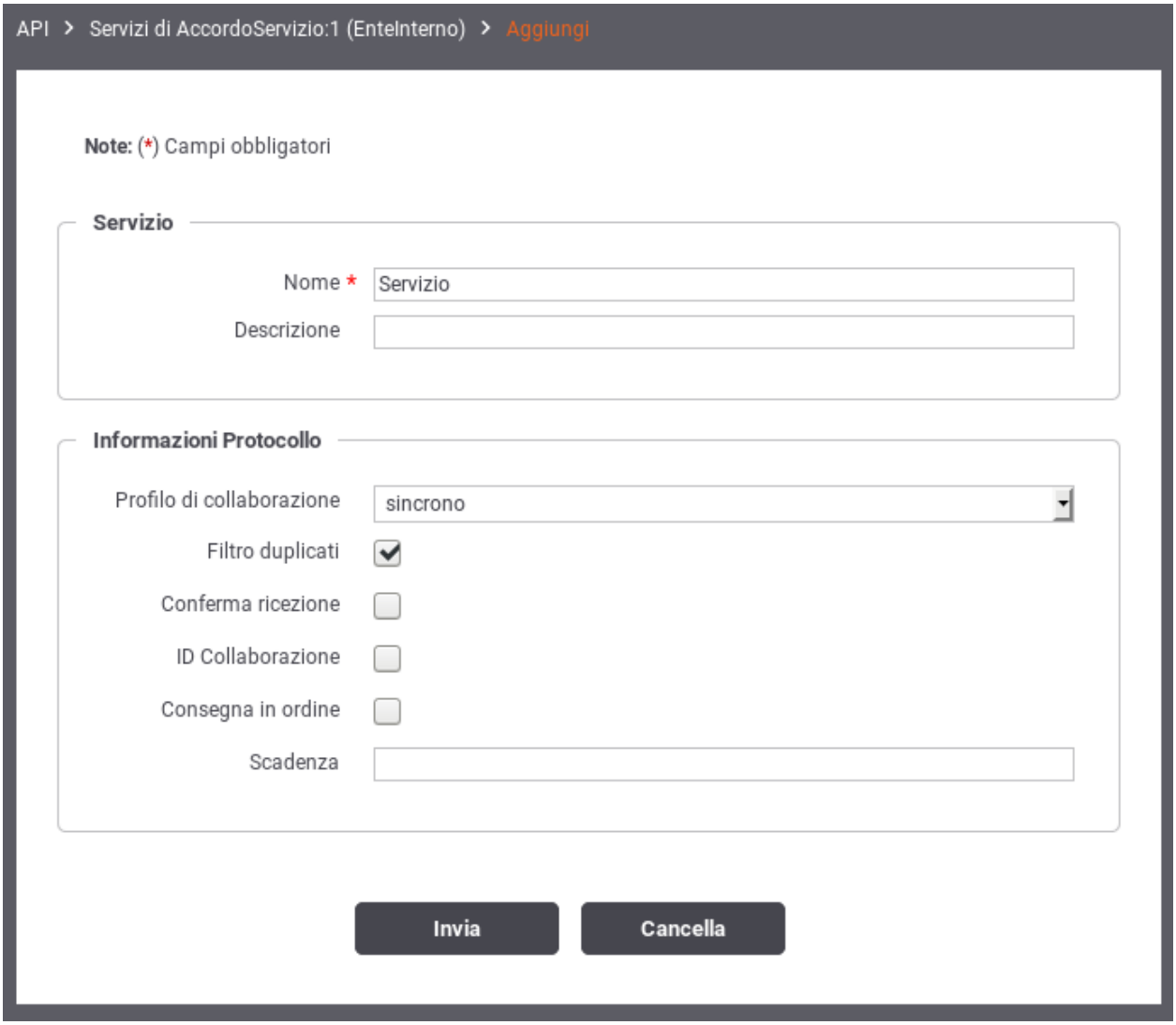


Fig. 5.2: Aggiunta Servizio SPCoop

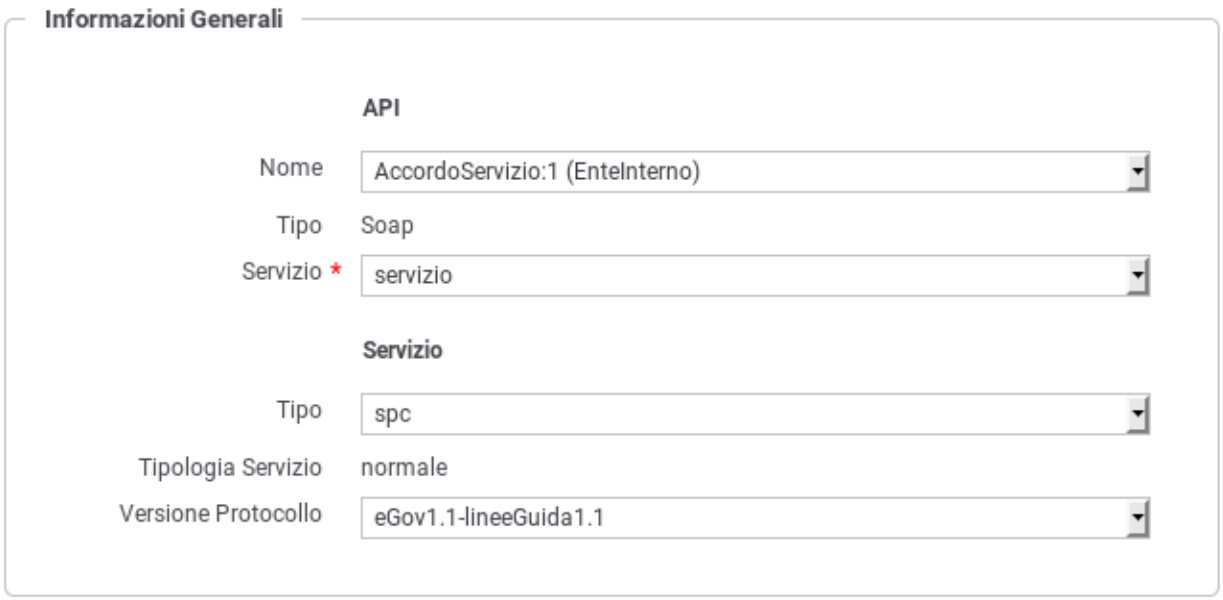


Fig. 5.3: Creazione erogazione SPCoop

5.2 Profili Asincroni

I servizi con profilo Asincrono richiedono alcuni passi di configurazione ulteriori rispetto alle normali configurazioni dei profili Sincroni e OneWay. Nelle sezioni successive vengono mostrati in dettagli i passi di configurazione ulteriori richiesti dal *Profilo di Collaborazione Asincrono Simmetrico* e dal *Profilo di Collaborazione Asincrono Asimmetrico*.

5.2.1 Profilo di Collaborazione Asincrono Simmetrico

La registrazione di un profilo asincrono simmetrico prevede che vengano correlati tra di loro due azioni di due servizi differenti presenti all'interno del solito accordo di servizio parte comune (API). Di seguito un esempio di tale configurazione.

Ruolo Fruitore

Per poter fruire di un servizio con il profilo asincrono simmetrico come prerequisito è richiesto almeno la registrazione di un applicativo client. La registrazione dell'applicativo fruitore deve prevedere, oltre alle normali configurazioni, la definizione di un connettore attraverso il quale la PdD consegnerà la risposta asincrona. È possibile definire un connettore per la "Risposta Asincrona" impostando la modalità avanzata nella console di gestione e accendo in modifica ad un applicativo client precedentemente registrato. Il link "Risposta Asincrona" consentirà di definire i parametri di accesso al backend per la gestione della risposta asincrona.

Una volta creato l'applicativo è possibile procedere con la registrazione della fruizione del servizio asincrono simmetrico dedicato all'invio della richiesta. Il controllo degli accessi della fruizione deve obbligatoriamente essere configurato per autenticare gli applicativi precedentemente registrati, in modo da identificare l'applicativo chiamante e poterlo associare alla sessione asincrona.

Terminata la registrazione della fruizione, dovrà essere registrata un'erogazione del servizio asincrono dedicato alla ricezione della risposta. La risposta ricevuta verrà consegnata al connettore definito per la risposta asincrona associato all'applicativo chiamante originario salvato nella sessione asincrona.

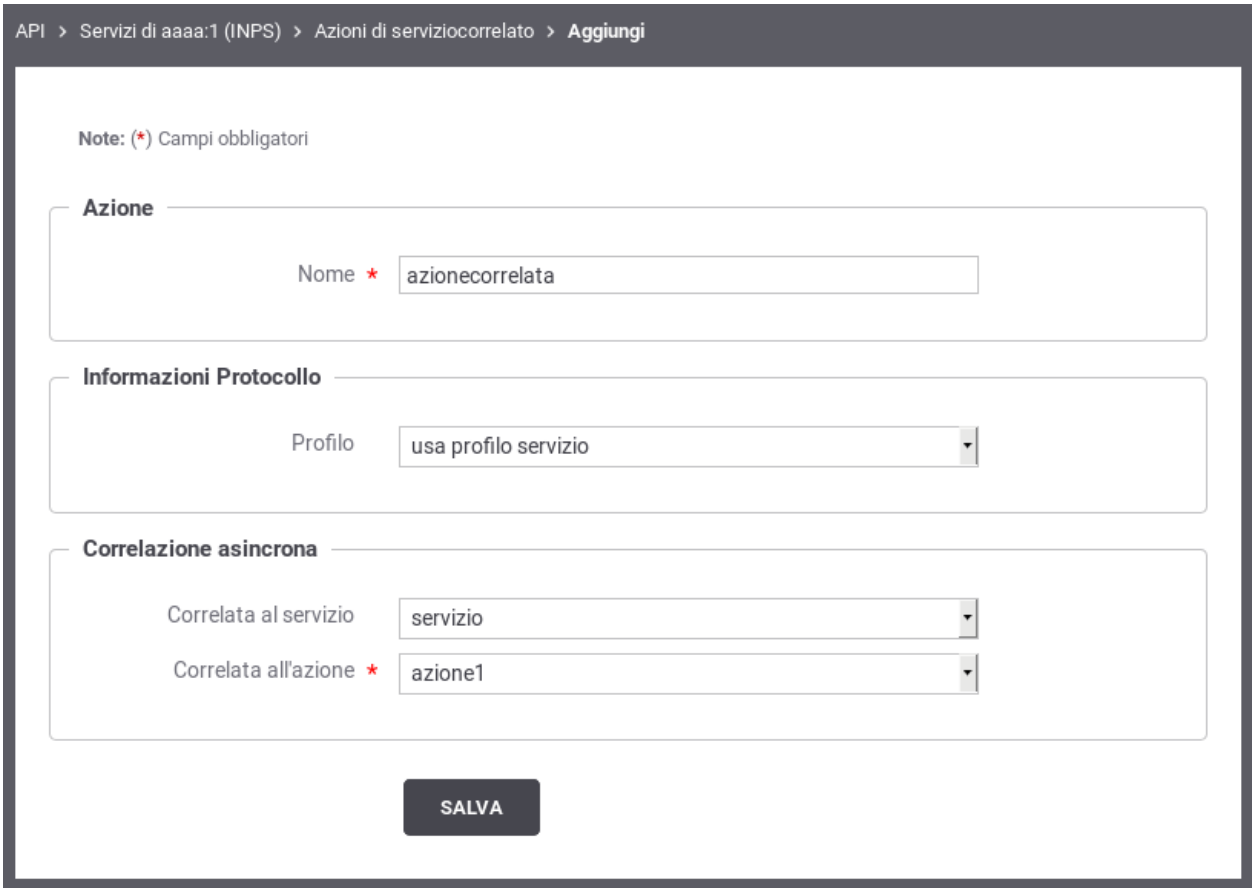


Fig. 5.4: Correlazione Asincrona Simmetrica

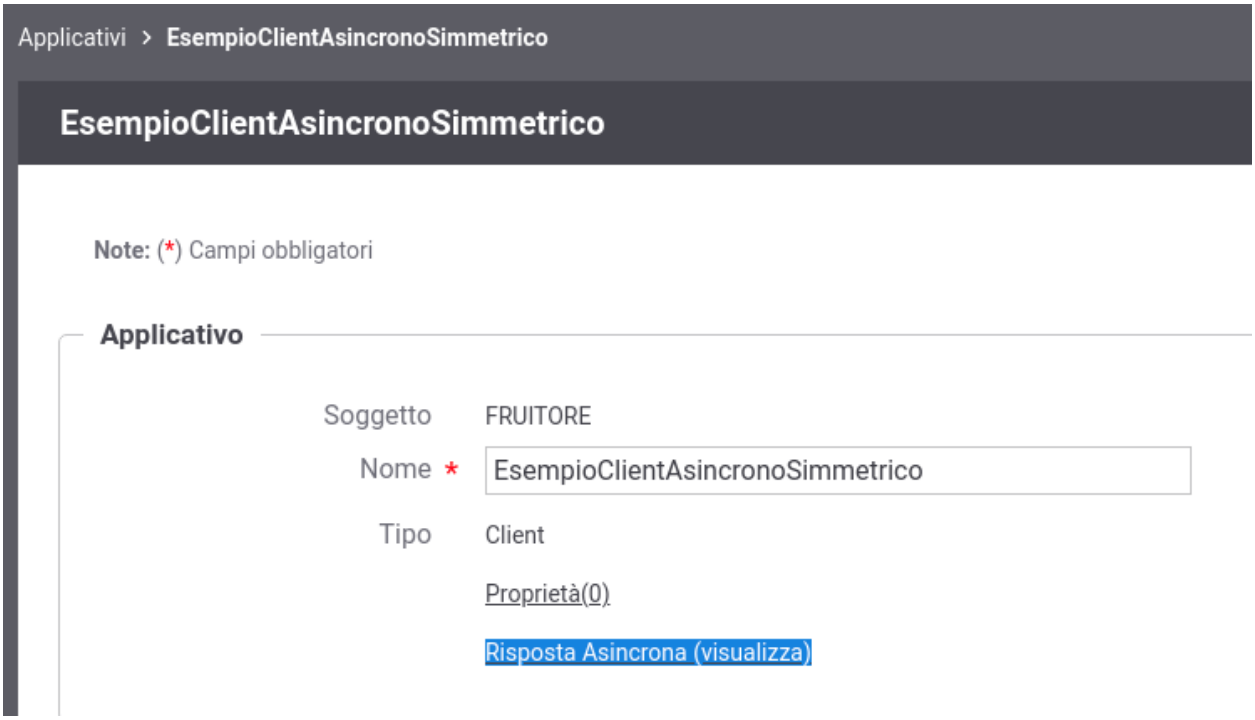


Fig. 5.5: Accesso alla configurazione dell'Applicativo client per la Risposta Asincrona

Nota: Configurazione Servizio Ricezione della Risposta

Durante la registrazione dell'erogazione del servizio di risposta, al connettore richiesto dalla maschera di configurazione può essere fornito un endpoint qualsiasi. Tale endpoint non verrà effettivamente utilizzato poichè la risposta asincrona ricevuta verrà inoltrata al backend configurato come "Risposta Asincrona" dell'applicativo client che ha effettuato la richiesta.

Ruolo Erogatore

Per poter erogare un servizio con il profilo asincrono simmetrico non sono richieste particolari configurazioni. Dovrà essere erogato il servizio relativo alla richiesta e fruito il servizio su cui inviare la risposta.

5.2.2 Profilo di Collaborazione Asincrono Asimmetrico

La registrazione di un profilo asincrono asimmetrico prevede che vengano correlati tra di loro due azioni, normalmente di uno stesso servizio, presenti all'interno dell'accordo di servizio parte comune (API). Di seguito un esempio di tale configurazione.

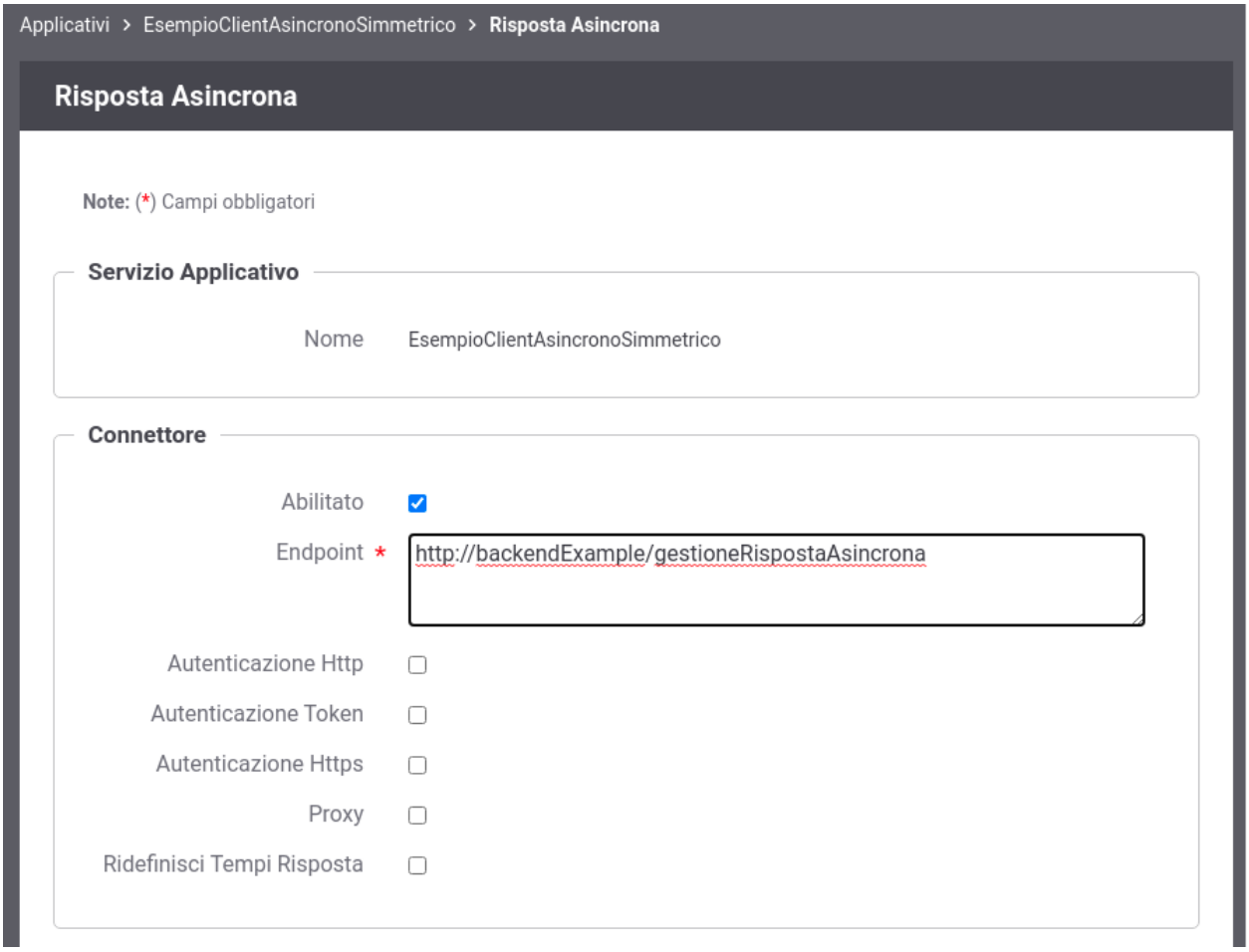


Fig. 5.6: Configurazione dell'Applicativo client per la Risposta Asincrona

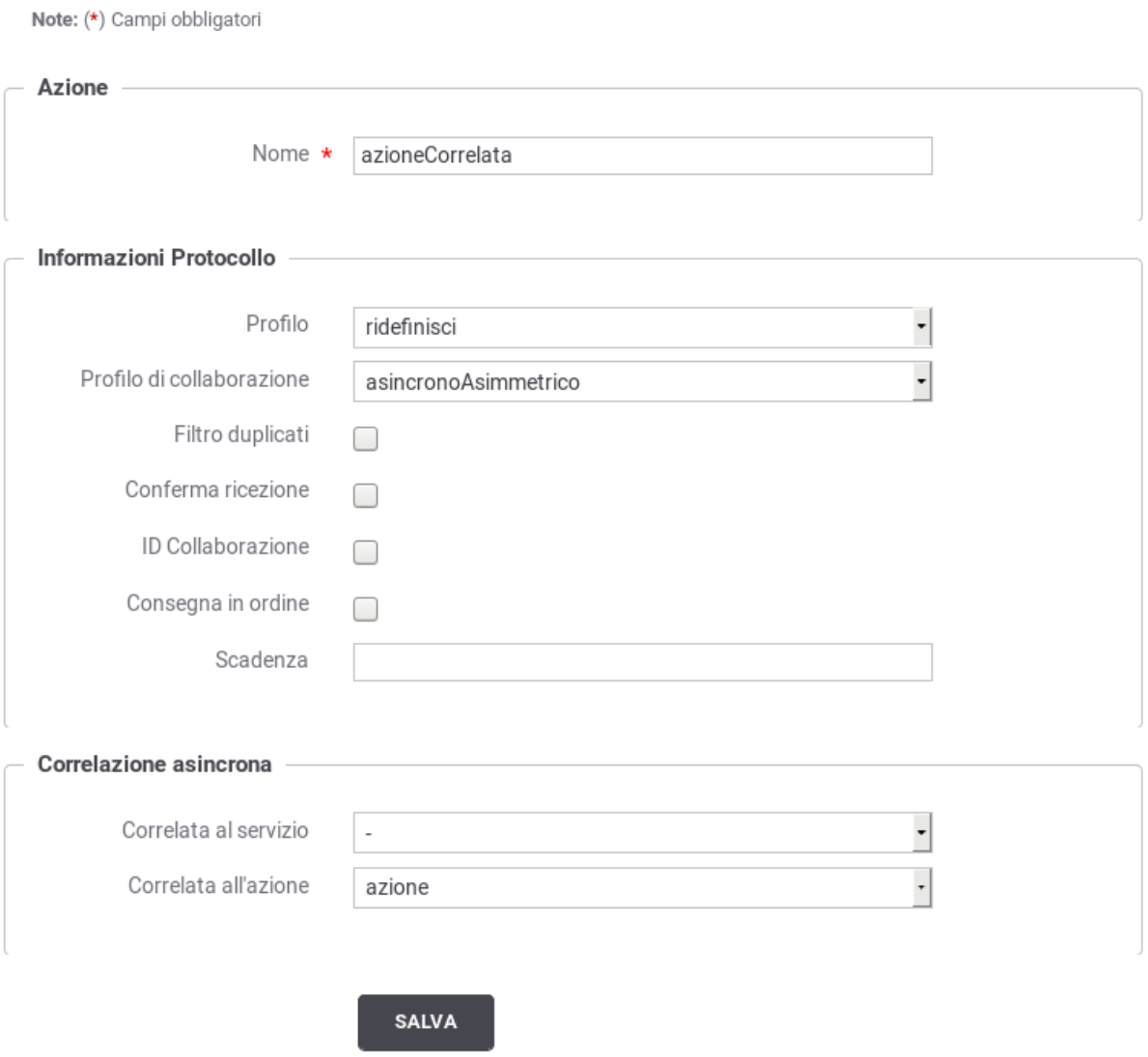


Fig. 5.7: Correlazione Asincrona Asimmetrica

Ruolo Fruitore

Per poter fruire un servizio con il profilo asincrono asimmetrico non sono richieste particolari configurazioni. Dovrà essere fruito il servizio su cui inviare la richiesta e richiedere l'esito della risposta.

Ruolo Erogatore

Per poter erogare un servizio con il profilo asincrono asimmetrico come prerequisito è richiesto la registrazione di un applicativo server. Durante la registrazione dell'applicativo possono essere indicati i parametri di accesso al backend a cui consegnare la richiesta asincrona. Una volta creato l'applicativo è possibile definire i parametri di accesso al backend a cui consegnare la richiesta stato asincrona impostando la modalità avanzata nella console di gestione ed entrando in modifica sull'applicativo server precedentemente registrato, dove sarà disponibile il link "Risposta Asincrona".

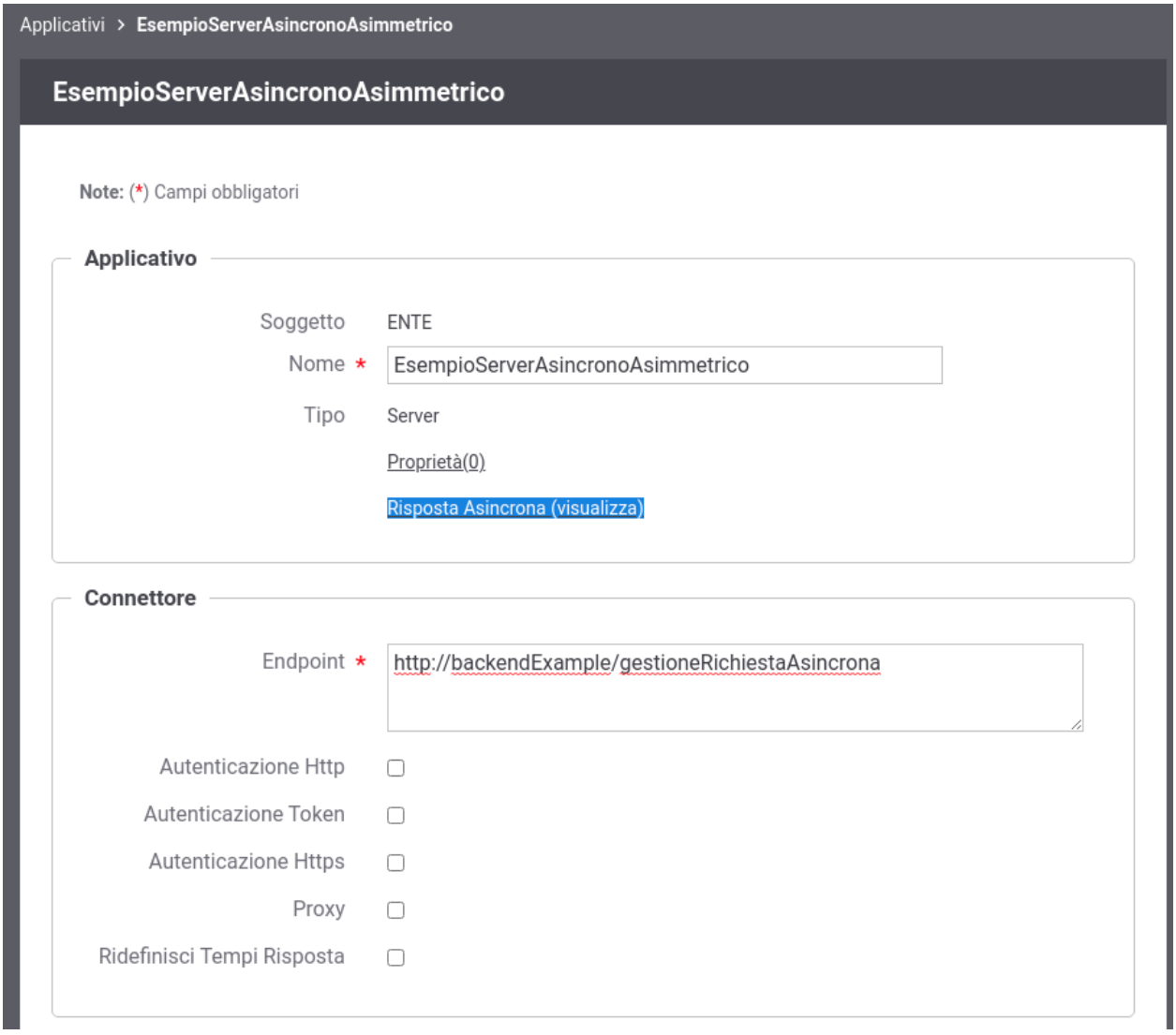


Fig. 5.8: Accesso alla configurazione dell'Applicativo server per la Risposta Asincrona

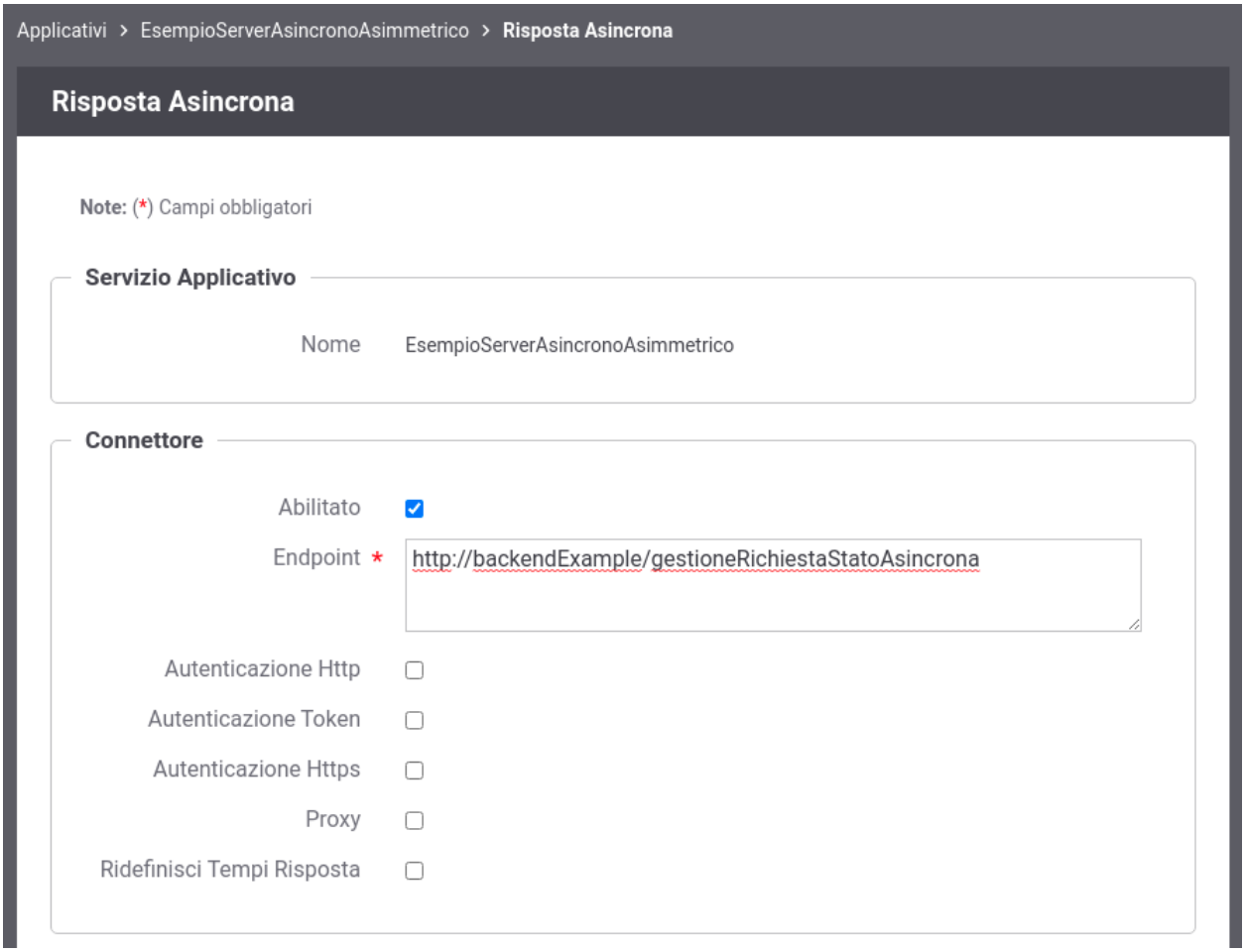


Fig. 5.9: Configurazione dell'Applicativo server per la Risposta Asincrona

Una volta creato l'applicativo è possibile procedere con la registrazione dell'erogazione del servizio con asincrono asimmetrico selezionando l'applicativo server precedentemente registrato come connettore di backend come mostrato in figura Fig. 5.10.

5.3 Interfacce WSDL (concettuale, logico ed implementativo)

La specifica SPCoop prevede che nell'accordo di servizio siano specificati i documenti WSDL del servizio applicativo erogatore e, nel caso di profili di collaborazione asincroni asimmetrici, anche quelli del servizio applicativo correlato erogato dal soggetto fruitore.

La Tabella 5.1 riepiloga i documenti necessari alla descrizione formale di un accordo di servizio che possono essere associati agli accordi parte comune e specifica se viene utilizzata la modalità avanzata della console

Tabella 5.1: Descrizione di un accordo di servizio

Nome Documento	Accordo
Specifica delle Interfacce	
WSDL Definitorio	Parte Comune
WSDL Concettuale	Parte Comune
WSDL Logico Erogatore	Parte Comune
WSDL Logico Fruitore	Parte Comune
Specifica delle Implementazioni	
WSDL Implementativo Erogatore	Parte Specifica
WSDL Implementativo Fruitore	Parte Specifica

5.4 Profili di gestione della busta eGov

L'interfaccia *completa* fornisce la possibilità di fruire/erogare di servizi SPCoop che non seguono le Linee Guida 1.1 ma si basano sul documento e-Gov 1.1. Questa funzionalità è utile sia per backward compatibility in quei domini dove i servizi non sono ancora stati adeguati al profilo descritto nelle Linee Guida 1.1, sia per usufruire di servizi infrastrutturali quali *consegna affidabile*, *consegna in ordine*, *conversazioni* che non sono presenti nel profilo Linee Guida 1.1.

Fruizione di un servizio.

Supponiamo di essere in un contesto dove vogliamo usufruire di un servizio erogato da un soggetto la cui PdD non è ancora stata adeguata a quanto descritto nelle Linee Guida 1.1. Per usufruire del servizio, il soggetto fruitore deve inviare buste conformi al profilo e-Gov 1.1, nonostante la propria porta di dominio sia già conforme alle Linee Guida 1.1. Per gestire tale contesto è possibile definire il soggetto erogatore con profilo *eGov1.1*. In un successivo momento, la PdD del soggetto erogatore può iniziare ad adeguarsi alle Linee Guida 1.1. Supponiamo che l'adeguamento sia incrementale, fornito per un servizio alla volta. Per usufruire dei servizi erogati da tale soggetto, con la giusta modalita (Linee Guida 1.1 o e-Gov 1.1) *è possibile ridefinire il profilo di gestione all'interno del servizio*.

Erogazione di un servizio.

Poniamoci in un contesto in cui la Porta di Dominio eroga dei servizi che rispettano quanto descritto nelle Linee Guida 1.1. In questo contesto, i soggetti di PdD che non si sono ancora adeguate alle linee guida, non potrebbero usufruire dei servizi. La PdD può essere configurata, in modo da erogare i servizi, per questi soggetti, secondo il profilo *eGov 1.1*. Questa configurazione richiede che al soggetto fruitore venga associato un profilo *eGov 1.1*. In un successivo momento, la PdD di un soggetto fruitore può iniziare ad adeguarsi alle Linee Guida 1.1. Si creano quindi due situazioni di transizione dove devono coesistere entrambe le specifiche:

- Un soggetto fruisce per alcuni servizi erogati secondo le specidiche e-Gov1.1, per altri secondo le Linee Guida 1.1

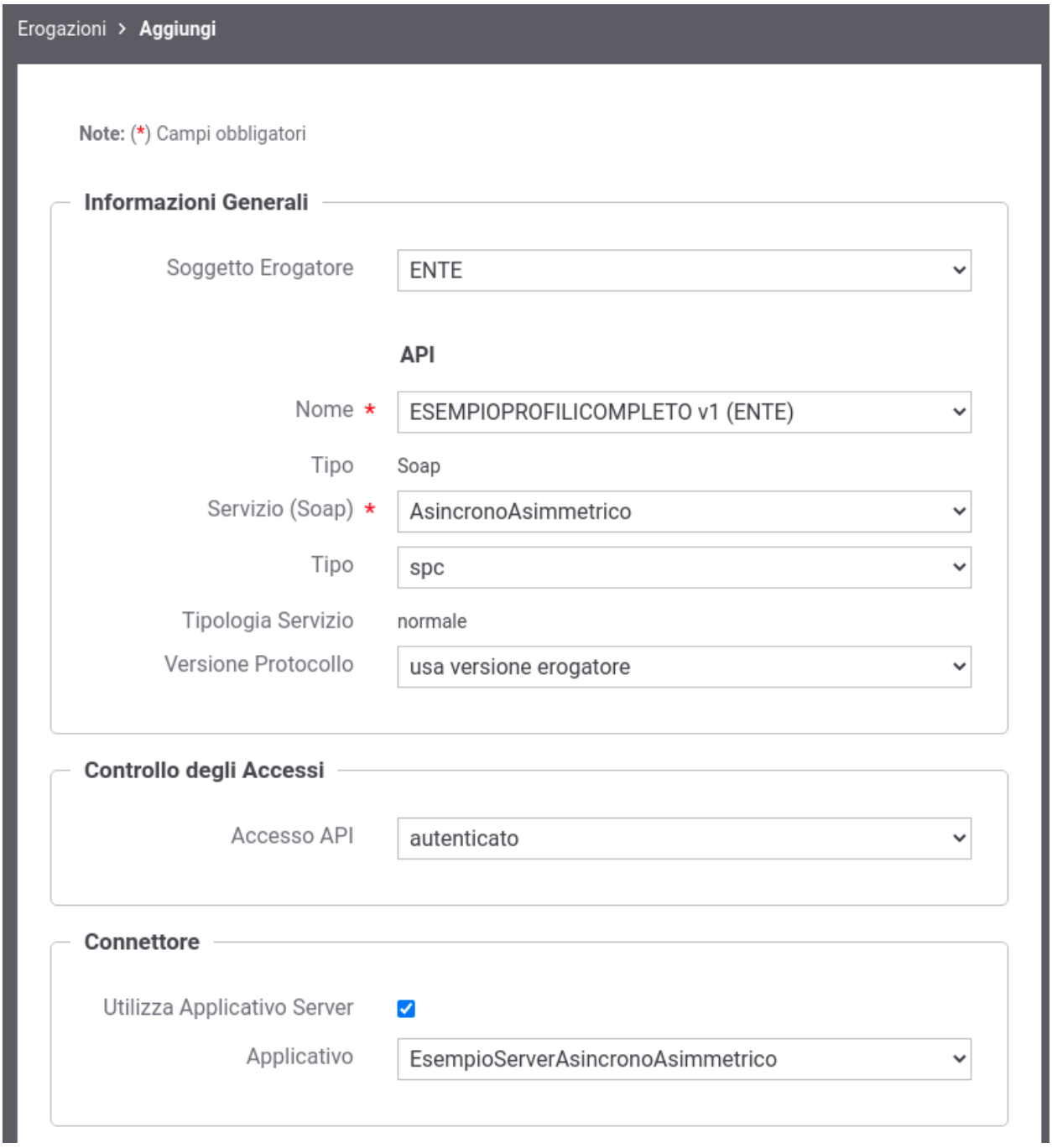


Fig. 5.10: Selezione dell'Applicativo server per l'erogazione del servizio con profilo asincrono asimmetrico.

- Uno o più fruitori accedono al un servizio erogato secondo le specidiche e-Gov1.1, altri secondo le Linee Guida 1.1

In entrmabi i casi, per erogare il servizio con la giusta modalita (linee guida o e-gov 1.1) è *possibile ridefinire il profilo di gestione impostandolo nella lista dei fruitori del servizio*.

5.4.1 Profilo di gestione e-Gov 1.1

Il documento delle linee guida ha deprecato alcune opzioni al fine di snellire la specifica. Per mantenere la conpatibilità con la vecchia versione viene sempre offerta la possibilità di specificare tali opzioni all'interno degli accordi di servizio. Tali funzionalità vengono impostate/validate all'interno della busta e-Gov solo se il servizio viene fruito/erogato con profilo *eGov1.1*.

Tabella 5.2: Opzioni della busta eGov

Nome	Default	Funzionalità
Filtro duplicati	true	Funzionalità di filtro delle buste duplicate (Imposta l'attributo inoltro del profilo di trasmissione al valore EGOV_IT_ALPIUUNAVOLTA).
Conferma Ricezione	false	Funzionalità di consegna affidabile delle buste spcoop attraverso l'utilizzo dei riscontri (Imposta l'attributo confermaRicezione del profilo di trasmissione al valore true).
ID Conversazione	false	Aggiunge un elemento Collaborazione alla busta (Diverse istanze di cooperazione possono essere correlate in un'unica conversazione).
Consegna in ordine	false	Consegna in ordine delle buste (Richiede Filtro Duplicati e Conferma Ricezione)
Scadenza		Assegna una scadenza temporale alla busta SPCoop

Di seguito un esempio di creazione di un accordo di servizio che richiede consegna affidabile tramite riscontri, filtro duplicati e id di conversazione per un servizio sincrono.

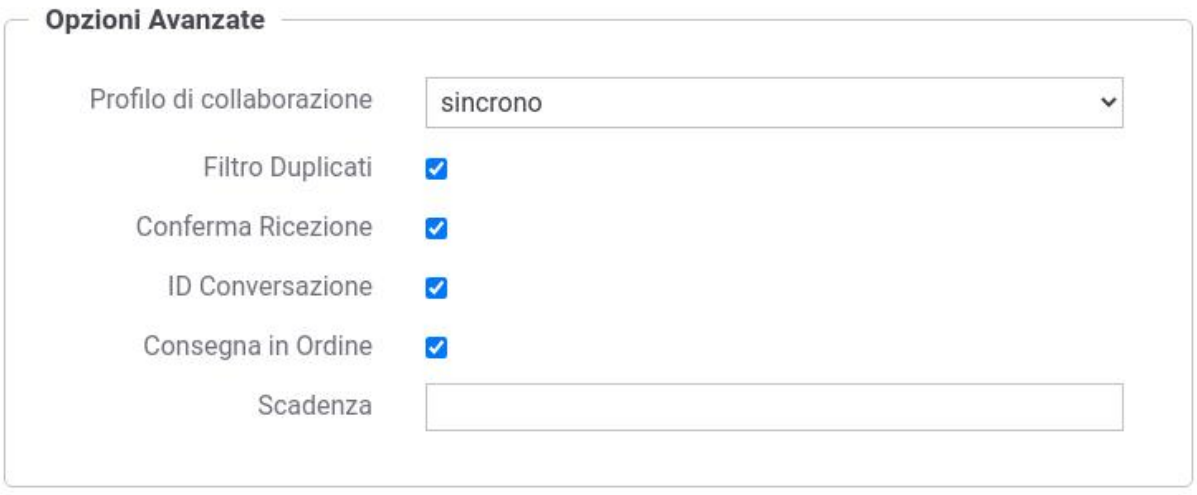


Fig. 5.11: Controlli avanzati sulle informazioni eGov relative all'accordo di servizio

CAPITOLO 6

Profilo “Fatturazione Elettronica”

Il profilo «Fatturazione Elettronica» consente di utilizzare GovWay come nodo di interconnessione al Sistema di Interscambio (SdI), responsabile della gestione dei flussi di fatturazione elettronica.

GovWay supporta la connessione al SdI attraverso lo scenario di interoperabilità su rete Internet basato sull'accesso al servizio *SdICoop*. Il servizio SdICoop prevede un protocollo di comunicazione, basato su SOAP, che veicola messaggi (fatture, archivi, notifiche e metadati) secondo la codifica dettata dalle specifiche tecniche (Per dettagli in merito si faccia riferimento alle Specifiche Tecniche SdI (https://www.fatturapa.gov.it/it/norme-e-regole/DocumentazioneSDI/).

Il profilo «Fatturazione Elettronica» consente, ai sistemi di gestione delle fatture di un ente, di non occuparsi della gestione del formato di scambio, previsto dal SdI, mantenendo un grado di interfacciamento notevolmente semplificato. Più in dettaglio:

- I gestionali dell'ente, registrati come applicativi su GovWay, inviano/ricevono le fatture e le notifiche, previste dal colloquio, nel formato originario XML senza ulteriori complessità.
- I metadati presenti nelle comunicazioni con il SdI vengono estratti ed elaborati da GovWay e trasmessi ai gestionali dell'ente tramite appositi *Header di Integrazione SdI*.
- La produzione dei metadati SdI, nel caso delle comunicazioni in uscita (fatturazione attiva), è a carico di GovWay che provvede anche a generare gli identificativi univoci da associare ai messaggi da trasmettere al SdI.

Per la produzione delle configurazioni necessarie a rendere operativo GovWay sono stati realizzati due wizard che guidano l'utente verso il corretto inserimento dei dati necessari. Gli scenari di configurazione supportati sono due e riguardano i casi della *Fatturazione Passiva* e *Fatturazione Attiva*.

6.1 Fatturazione Passiva

Nello scenario di fatturazione passiva si utilizza GovWay per la ricezione delle fatture in arrivo dal SdI. GovWay attua la decodifica del messaggio SdI ricevuto, al fine di estrarre i file fattura in esso contenuti e trasmetterli, nel formato FatturaPA, all'applicativo registrato come destinatario.

Lo scenario complessivo, relativo alla Fatturazione Passiva, è quello illustrato in Fig. 6.1.

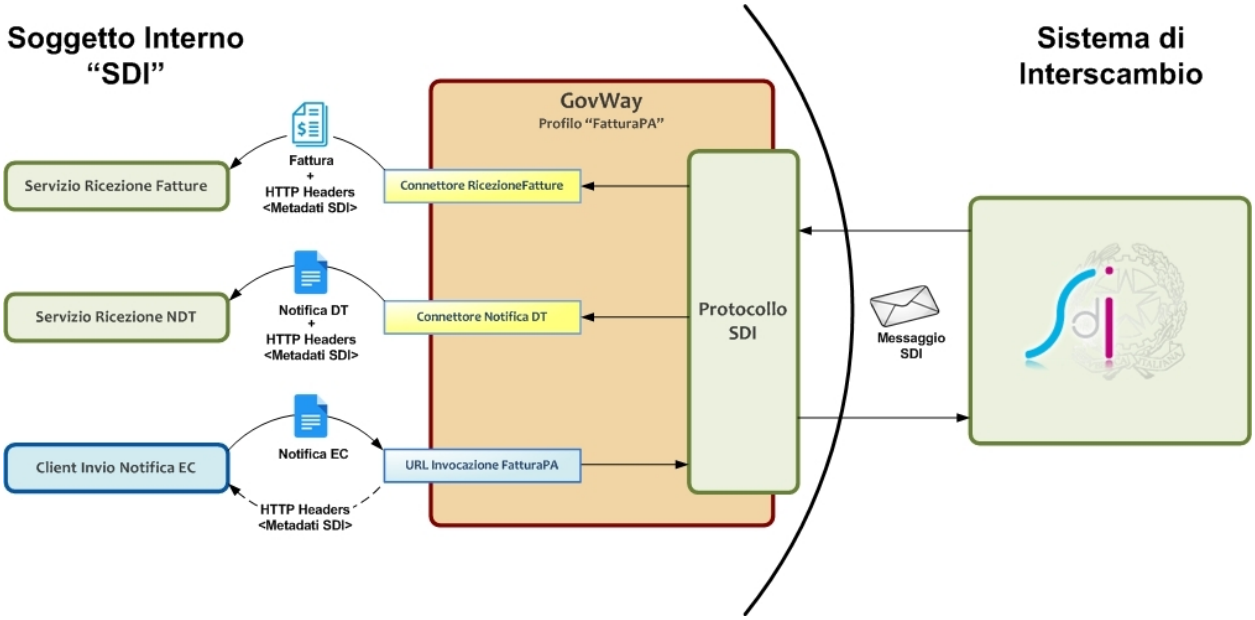


Fig. 6.1: Scenario di interoperabilità relativo alla Fatturazione Passiva

Descriviamo per punti i passi significativi di questo scenario:

- *Servizio Ricezione Fatture*. Per consentire a GovWay di consegnare le fatture ricevute dal SdI è necessario esporre un servizio i cui riferimenti per l'accesso dovranno essere configurati nel contesto del *Connettore RicezioneFatture*, presente nella configurazione di GovWay.

  Le fatture vengono ricevute da GovWay formato codificato dal protocollo SdI, e comprendono il lotto delle fatture, con i relativi allegati, e un insieme di metadati che descrivono il contesto di invocazione. GovWay si occupa di estrarre le informazioni presenti, elaborando il messaggio SdI, provvedendo quindi a consegnare il lotto di fatture al servizio destinatario, nel formato *FatturaPA* attraverso l'invocazione di una HTTP POST. I metadati raccolti dal messaggio SdI vengono forniti, nel contesto della medesima richiesta, sotto forma di HTTP Headers (fare riferimento alla Tabella 6.1).

Nota: Nella configurazione di default GovWay non consegna il file Metadati all'applicativo. È possibile attivare la consegna abilitando la proprietà "org.openspcoop2.protocol.sdi.fatturazionePassiva.consegnaFileMetadati" all'interno del file /etc/govway/sdi_local.properties. Il file Metadati verrà consegnato, codificato in base64, nell'header HTTP "GovWay-SDI-FileMetadati".

- *Client Invio Notifica EC*. I sistemi dell'ente, dopo aver ricevuto le fatture, inviano le *Notifiche di Esito Committente*, previste dal protocollo SdI, utilizzando un apposito servizio di GovWay. La URL di invocazione di tale servizio sarà disponibile al termine del processo di configurazione descritto più avanti. GovWay provvede a codificare il messaggio SdI di richiesta contenente il messaggio di notifica ricevuto dall'applicativo mittente.

I metadati prodotti per il messaggio SdI, unitamente all'identificativo messaggio univoco generato, vengono restituiti all'applicativo mittente sotto forma di HTTP Headers (fare riferimento alla Tabella 6.2).

- *Servizio Ricezione NDT*. Per consentire a GovWay di consegnare le eventuali *Notifiche di Decorrenza Termini* è necessario esporre un servizio i cui riferimenti per l'accesso dovranno essere configurati nel contesto del *Connettore NotificaDT*, presente nella configurazione di GovWay.

  GovWay consegna le notifiche DT nel formato originale tramite una HTTP POST, includendo come HTTP Headers i metadati estratti dal messaggio SdI originariamente ricevuto (fare riferimento alla Tabella 6.3).

Tabella 6.1: Header di Integrazione Ricezione Fattura

Header	Descrizione
GovWay-SDI-FormatoArchivioBase64	Indica se il file fattura è codificato in formato Base64
GovWay-SDI-FormatoArchivioInvioFattura	Indica se è stata utilizzata la modalità di firma CAdES o XAdES (P7M o XML)
GovWay-SDI-FormatoFatturaPA	Indice di versione del formato FatturaPA adottato
GovWay-SDI-IdentificativoSdI	Identificativo assegnato dal SdI alla fattura
GovWay-SDI-MessageId	Identificativo assegnato alla fattura dall'ente trasmittente
GovWay-SDI-NomeFile	Nome del file fattura
GovWay-SDI-NomeFileMetadati	Nome del file di metadati
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Tabella 6.2: Header di Integrazione Invio Notifica EC

Header	Descrizione
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Tabella 6.3: Header di Integrazione Ricezione Notifica DT

Header	Descrizione
GovWay-SDI-IdentificativoSdI	Identificativo assegnato dal SdI alla fattura
GovWay-SDI-NomeFile	Nome del file fattura
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Per produrre le configurazioni necessarie all'utilizzo dello scenario di fatturazione passiva, è possibile utilizzare il wizard messo a disposizione per semplificare l'attività di configurazione di GovWay. I passi da eseguire sono i seguenti:

1. Scaricare il govlet per la fatturazione passiva al seguente indirizzo:
   - http://www.govway.org/govlets/fatturazione-passiva.zip
2. Avviare il govlet posizionandosi sulla sezione *Configurazione* > *Importa* della GovWayConsole e selezionare il file appena scaricato come oggetto da importare.
3. *Soggetto SDI*: al primo step del wizard viene richiesto di indicare, tra gli elementi presenti nella lista a discesa, il soggetto interno destinatario delle fatture. Si tratta di un soggetto appartenente al profilo «FatturaPA».
4. *Servizio SdIRiceviNotifica erogato dal Sistema di Interscambio*: al secondo step viene richiesto di indicare la URL che corrisponde all'endpoint del servizio SdIRiceviNotifica, necessario per l'invio delle *Notifiche di Esito Committente*.

Nota: il valore suggerito dalla maschera di configurazione del govlet fa riferimento alla url del sistema di produzione SDI. Se si vuole configurare un servizio di test è necessario cambiare tale valore ed impostare il riferimento all'ambi-

ente di test SDI. I certificati, sia per l'ambiente di test che di produzione, devono essere stati inseriti nel truststore di GovWay.

5. *Credenziali per accesso URL NotificaEsito*: al terzo step viene richiesto di fornire il criterio di autenticazione utilizzato dall'applicativo per inviare la notifica di esito committente.
6. *Applicativo per consegna FatturaPA*: al quarto step viene richiesto di fornire i dati di configurazione del connettore, utilizzato da GovWay per la consegna delle fatture. La configurazione del connettore comprende: endpoint, credenziali di autenticazione ed eventualmente i riferimenti del proxy.
7. *Applicativo per consegna NotificaDecorrenzaTermini*: al quinto ed ultimo step viene richiesto di fornire i dati di configurazione del connettore, utilizzato da GovWay per la consegna della notifica di decorrenza termini. La configurazione del connettore comprende: endpoint, credenziali di autenticazione ed eventualmente i riferimenti del proxy.

6.1.1 Ricezione Fatture e Notifiche di Decorrenza Termini

Allo SdI, per raggiunge il servizio di RicezioneFatture su Govway, dovrà essere comunicata la seguente URL:

```
https://<host-govway</govway/sdi/in/<SoggettoSDI>/RicezioneFatture/v1
```

Le fatture e le notifiche saranno consegnati all'applicativo dell'ente secondo le modalità specificate durante l'esecuzione del Govlet. In fase di consegna delle fatture e delle notifiche verranno generati rispettivamente gli header descritti nelle tabelle precedenti.

6.1.2 Invio della Notifica di Esito Committente

Per l'invio della Notifica di Esito Committente l'applicativo deve utilizzare:

- Una URL così composta:

  ```
  http://<host-govway>/govway/sdi/out/xml2soap/<SoggettoSDI>/CentroServiziFatturaPA/
  ↪SdIRiceviNotifica/v1?NomeFile=<NomeFileFattura>&IdentificativoSdI=
  ↪<identificativoSDI>
  ```

  dove:

  - *host-govway*: è l'hostname con cui è raggiungibile l'istanza di Govway.
  - *SoggettoSDI*: il soggetto interno destinatario delle fatture, come configurato durante l'esecuzione del govlet di fatturazione passiva.
  - *NomeFileFattura*: è il nome del file che contiene la fattura cui fa riferimento la notifica EC.
  - *identificativoSDI*: è l'identificativo SDI che fa riferimento al lotto della fattura ricevuta.

- L'invocazione deve essere corredata dalle credenziali che sono state indicate durante la configurazione tramite il relativo govlet.
- Utilizzare l'header http *Content-Type* valorizzato con *text/xml* o *application/xml*

Un esempio di invio di una fattura viene fornito tramite il seguente comando curl:

Soggetto Interno al Dominio

In questo esempio si suppone che il nome del soggetto (riferito precedentemente come *SoggettoSDI*) fornito durante la fase di installazione di GovWay sia *Ente*.

```
curl -X POST -basic --user SdIRiceviNotifica:123456 \
--data-binary @IT01234567890_11111_EC_001.xml \
-H "Content-Type: application/xml" \
"http://127.0.0.1:8080/govway/sdi/out/xml2soap/Ente/CentroServiziFatturaPA/
↪SdIRiceviNotifica/v1?NomeFile=IT01234567890_11111.xml&IdentificativoSdI=345"
```

Nota: La generazione di un nome di file univoco da associare alla notifica di esito committente viene gestita da GovWay.

È possibile disabilitare tale gestione disabilitando la proprietà "org.openspcoop2.protocol.sdi.fatturazionePassiva.nomeFile.gestione" nel file "/etc/govway/sdi_local.properties". Se viene disabilitata la funzionalità (attiva per default), la gestione dei nomi dei file (correttezza sintattica, univocità, ...) è demandata all'Applicativo Client che deve obbligatoriamente fornire il nome da associare alla notifica di esito committente del file attraverso uno dei seguenti modi:

- query parameter "NomeFile"
- header http "SDI-NomeFile"
- header http "GovWay-SDI-NomeFile"

6.2 Fatturazione Attiva

Nello scenario di fatturazione attiva si utilizza GovWay per l'invio delle fatture al SdI. GovWay attua la codifica dei file ricevuti al fine di produrre un messaggio valido per l'invio al SdI.

Lo scenario complessivo, relativo alla Fatturazione Attiva, è quello illustrato in Fig. 6.2.

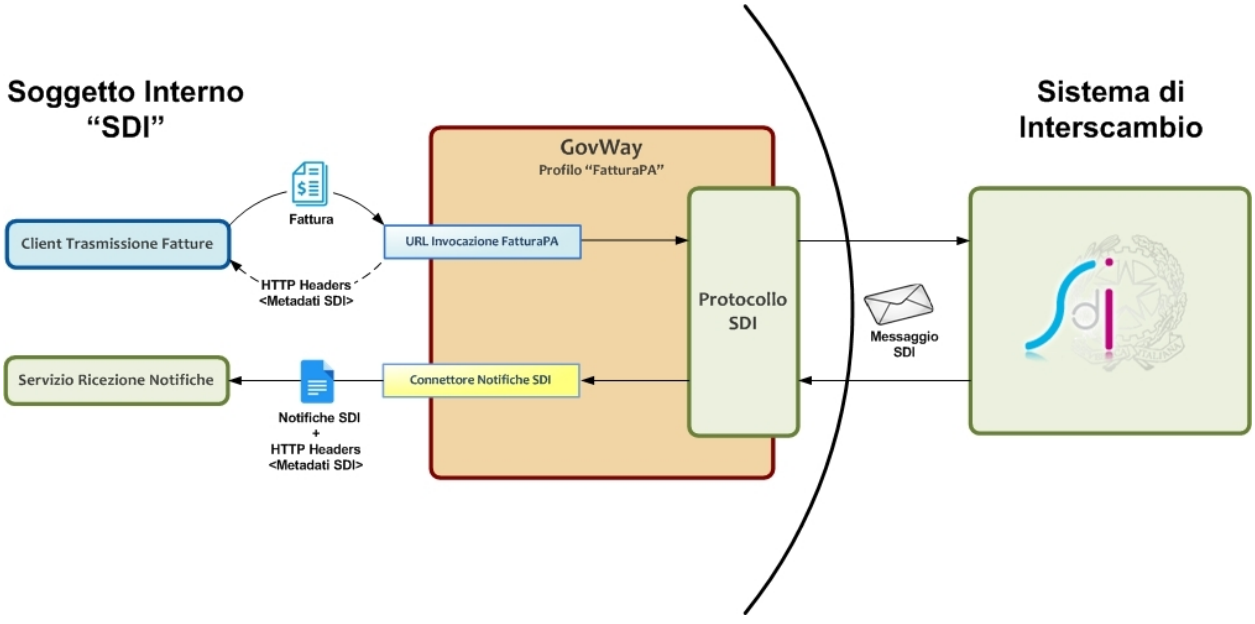


Fig. 6.2: Scenario di interoperabilità relativo alla Fatturazione Attiva

Descriviamo per punti i passi significativi di questo scenario:

- *Client Trasmissione Fatture*. I sistemi dell'ente possono trasmettere le fatture al SdI tramite un apposito servizio di ricezione di GovWay. La URL di invocazione di tale servizio sarà disponibile al termine del processo di configurazione dello scenario di fatturazione attiva descritto più avanti. Una volta ricevuta la fattura, nel formato previsto da FatturaPA, GovWay provvede a codificare il messaggio SdI di richiesta contenente la fattura da trasmettere. I metadati prodotti per il messaggio SdI, unitamente all'identificativo SdI, vengono restituiti all'applicativo mittente sotto forma di HTTP Headers (fare riferimento alla Tabella 6.4).
- *Servizio Ricezione Notifiche*. I sistemi dell'ente devono esporre un servizio adibilito alla ricezione delle notifiche che il SdI invia successivamente alla trasmissione di una fattura. I riferimenti per l'accesso a tale servizio dovranno essere configurati nel contesto del *Connettore NotificheSDI*, presente nella configurazione di GovWay.

  GovWay consegna le notifiche, al servizio dell'ente, nel formato originale tramite una HTTP POST, includendo come HTTP Headers i metadati estratti dal messaggio SdI originariamente ricevuto (fare riferimento alla Tabella 6.5).

Tabella 6.4: Header di Integrazione "Trasmissione Fatture"

Header	Descrizione
GovWay-SDI-IdentificativoSdI	Identificativo assegnato dal SdI alla fattura
GovWay-SDI-NomeFile	Nome del file fattura
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Tabella 6.5: Header di Integrazione "Ricezione Notifiche"

Header	Descrizione
GovWay-SDI-IdentificativoSdI	Identificativo assegnato dal SdI alla fattura
GovWay-SDI-NomeFile	Nome del file fattura
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Per produrre le configurazioni necessarie all'utilizzo dello scenario di fatturazione attiva, è possibile utilizzare il wizard messo a disposizione per semplificare l'attività di configurazione di GovWay. I passi da eseguire sono i seguenti:

1. Scaricare il govlet per la fatturazione attiva al seguente indirizzo
   - http://www.govway.org/govlets/fatturazione-attiva.zip
2. Avviare il govlet posizionandosi sulla sezione *Configurazione* > *Importa* della GovWayConsole e selezionare il file appena scaricato come oggetto da importare.
3. *Soggetto SDI*: al primo step viene richiesto di indicare, tra gli elementi presenti nella lista a discesa, il soggetto interno mittente delle fatture. Si tratta di un soggetto appartenente al profilo «FatturaPA».
4. *Servizio SdIRiceviFile erogato dal Sistema di Interscambio*: al secondo step viene richiesto di indicare la URL che corrisponde all'endpoint del servizio SdIRiceviFile, erogato dal SdI per la trasmissione delle fatture.

Nota: il valore suggerito dalla maschera di configurazione del govlet fa riferimento alla url del sistema di produzione SDI. Se si vuole configurare un servizio di test è necessario cambiare tale valore ed impostare il riferimento all'ambiente di test SDI. I certificati, sia per l'ambiente di test che di produzione, devono essere stati inseriti nel truststore di GovWay.

---

5. *Credenziali per accesso URL RiceviFile*: al terzo step viene richiesto di fornire il criterio di autenticazione utilizzato dall'applicativo per invocare la url del GovWay per la trasmissione delle fatture.
6. *Applicativo per consegna Notifiche*: al quarto ed ultimo step viene richiesto di fornire i dati di configurazione del connettore, utilizzato da GovWay per la consegna delle notifiche inviate dal SdI, successivamente alla trasmissione di una determinata fattura. La configurazione del connettore comprende: endpoint, credenziali di autenticazione ed eventualmente i riferimenti del proxy.

6.2.1 Invio della fattura

Per l'invio della fattura l'applicativo mittente deve utilizzare:

- Una URL così composta:

```
http://<host-govway>/govway/sdi/out/xml2soap/<SoggettoSDI>/CentroServiziFatturaPA/
↪SdIRiceviFile/v1?Versione=<VersioneFatturaPA>&TipoFile=<TipoFile>&IdPaese=
↪<IdPaese>&IdCodice=<IdCodice>
```

  dove:

  - *host-govway*: è l'hostname con cui è raggiungibile l'istanza di Govway.
  - *SoggettoSDI*: il soggetto interno al dominio come configurato durante l'esecuzione del govlet di fatturazione passiva.
  - *Versione*: versione della fattura che si sta inviando: FPA12 (Fattura 1.2 per Pubbliche Amministrazione), FPR12 (Fattura 1.2 per Privati), SDI11 e SDI10 (Fattura per Pubblica amministrazione versione 1.1. e 1.0).
  - *TipoFile*: tipo di fattura: XML (Fattura firmata XADES), P7M (Fattura firmata CADES) o ZIP (archivio di fatture).
  - *IdPaese e IdCodice*: dati del trasmittente della fattura.

- L'invocazione deve essere corredata dalle credenziali che sono state indicate durante la configurazione tramite il relativo govlet.
- A seconda del tipo di fattura deve essere utilizzato il corretto header http *Content-Type*:
  - XML: è possibile utilizzare *text/xml* o *application/xml*
  - P7M: *application/pkcs7-mime*
  - XML: *application/zip*

Un esempio di invio di una fattura viene fornito tramite il seguente comando curl:

Soggetto Interno al Dominio

In questo esempio si suppone che il nome del soggetto (riferito precedentemente come *SoggettoSDI*) fornito durante la fase di installazione di GovWay sia *Ente*.

```
curl -X POST -basic --user SdIRiceviFile:123456 \
--data-binary @IT01234567890_11111.xml.p7m \
-H "Content-Type: application/pkcs7-mime" \
"http://127.0.0.1:8080/govway/sdi/out/xml2soap/Ente/CentroServiziFatturaPA/
↪SdIRiceviFile/v1?Versione=SDI10&TipoFile=P7M&IdPaese=IT&IdCodice=01629370097"
```

Nota: La generazione di un nome di file univoco da associare alla fattura viene gestita da GovWay.

È possibile disabilitare tale gestione disabilitando la proprietà "org.openspcoop2.protocol.sdi.fatturazioneAttiva.nomeFile.gestione" nel file "/etc/govway/sdi_local.properties". Se viene disabilitata la funzionalità (attiva per default), la gestione dei nomi dei file (correttezza sintattica, univocità, ...) è demandata all'Applicativo Client che deve obbligatoriamente fornire il nome del file da associare alla fattura attraverso uno dei seguenti modi:

- query parameter "NomeFile"
- header http "SDI-NomeFile"
- header http "GovWay-SDI-NomeFile"

6.2.2 Ricezione delle Notifiche dallo Sdi

Allo Sdi dovrà essere comunicata la seguente url che utilizzerà per inoltrare le notifiche:

```
https://<host-govway</govway/sdi/in/<SoggettoSDI>/TrasmissioneFatture/v1
```

Le notifiche ricevute verranno consegnate secondo le modalità specificate durante l'esecuzione del Govlet. In fase di consegna verranno generati gli header descritti nella Tabella 6.5

CAPITOLO 7

Strumenti

7.1 Runtime

Questa sezione consente di visualizzare dati in tempo reale relativi al contesto di esecuzione del gateway, con la possibilità di effettuare alcune modifiche di stato. Le informazioni presenti sono:

- *Runtime*:
  - *Download*: consente di effettuare il download di un file di testo che contiene tutti i parametri visualizzati nella pagina.
  - *ResetAllCaches*: consente di effettuare il reset contemporaneo di tutte le cache utilizzate dal gateway.
- *Informazioni Generali*: Informazioni sul prodotto e sul software di base.
- *Stato Servizi*: Consente di abilitare/disabilitare in tempo reale i servizi per l'elaborazione delle richieste in ingresso intra ed extra dominio.
- *Informazioni Diagnostica*: Riferimenti ai file di log attivi per il prodotto, con la possibilità di modificare in tempo reale il livello di verbosità degli stessi.
- *Informazioni Tracciamento*: Riferimenti ai file contenenti il tracciamento delle richieste in elaborazione sul gateway, con la possibilità di abilitare/disabilitare le specifiche fonti.
- *Informazioni Database*: Informazioni relative la piattaforma database adottata.
- *Informazioni SSL*: Informazioni sulla configurazione SSL.
- *Informazioni Internazionalizzazione*: Informazioni sulla configurazione del servizio di internazionalizzazione.
- *Informazioni Timezone*: Timezone attivo.
- *Informazioni Java Networking*: Parametri di configurazione inerenti la configurazione del networking a livello Java.
- *Informazioni Modalità Gateway*: Contesti configurati per ciascuna specifica modalità operativa.
- *Cache*: Parametri di configurazione di tutte le cache adottate dal gateway, con la possibilità di effettuare il reset di ciascuna singolarmente.

- *Connessioni Attive*: Evidenza in tempo reale delle connessioni attive verso altri software a supporto (database, broker jms, ecc.)
- *Transazioni Attive*: Riferimenti alle transazioni in corso di elaborazione.
- *Connessioni HTTP Attive*: Evidenza in tempo reale delle connessioni HTTP, aperte in uscita, per l'elaborazione delle richieste in corso.

7.2 Auditing

La funzionalità di *auditing* consente di tracciare il comportamento degli utenti della govwayConsole, al fine di verificare le operazioni eseguite e i loro effetti.

Per gli aspetti di configurazione della funzionalità di auditing si rimanda alla sezione *Auditing*.

In questa sezione descriviamo le interfacce della govwayConsole dedicate alla consultazione delle informazioni raccolte tramite il servizio di auditing.

Gli utenti della govwayConsole aventi il permesso [A] Auditing (vedi *Utenti*) hanno accesso alla funzionalità di consultazione dei dati presenti nel repository del servizio di auditing.

Per accedere al servizio di consultazione selezionare la voce **Auditing** nella sezione **Reportistica** del menu laterale sinistro. La consultazione dei dati di auditing avviene tramite ricerche effettuate impostando i criteri attraverso il form riportato in Fig. 7.1.

Vediamo adesso il significato dei parametri per la ricerca dei dati di auditing:

- *Criteri di Ricerca*
  - **Inizio Intervallo**: Data iniziale che serve ad impostare l'intervallo temporale su cui restringere la ricerca dei dati di auditing. Lasciare il campo vuoto equivale all'impostazione *illimitato*.
  - **Fine Intervallo**: Data finale che serve ad impostare l'intervallo temporale su cui restringere la ricerca dei dati di auditing. Lasciare il campo vuoto equivale all'impostazione *illimitato*.
  - **Utente**: Consente di restringere la ricerca alle sole operazioni effettuate da un determinato utente. Il campo lasciato vuoto equivale a *qualasiasi utente*.
- *Operazione*
  - **Tipo**: Filtro per tipo di operazione, distinguendo tra:
    - *ADD*: creazione di un'entità
    - *CHANGE*: modifica di un'entità
    - *DEL*: cancellazione di un'entità
    - *LOGIN*: accesso alla govwayConsole
    - *LOGOUT*: disconnessione dalla govwayConsole
  - **Stato**: Filtro in base allo stato dell'operazione, distinguendo tra:
    - *requesting*: in fase di richiesta
    - *error*: terminata con errore
    - *completed*: terminata correttamente
- *Oggetto*

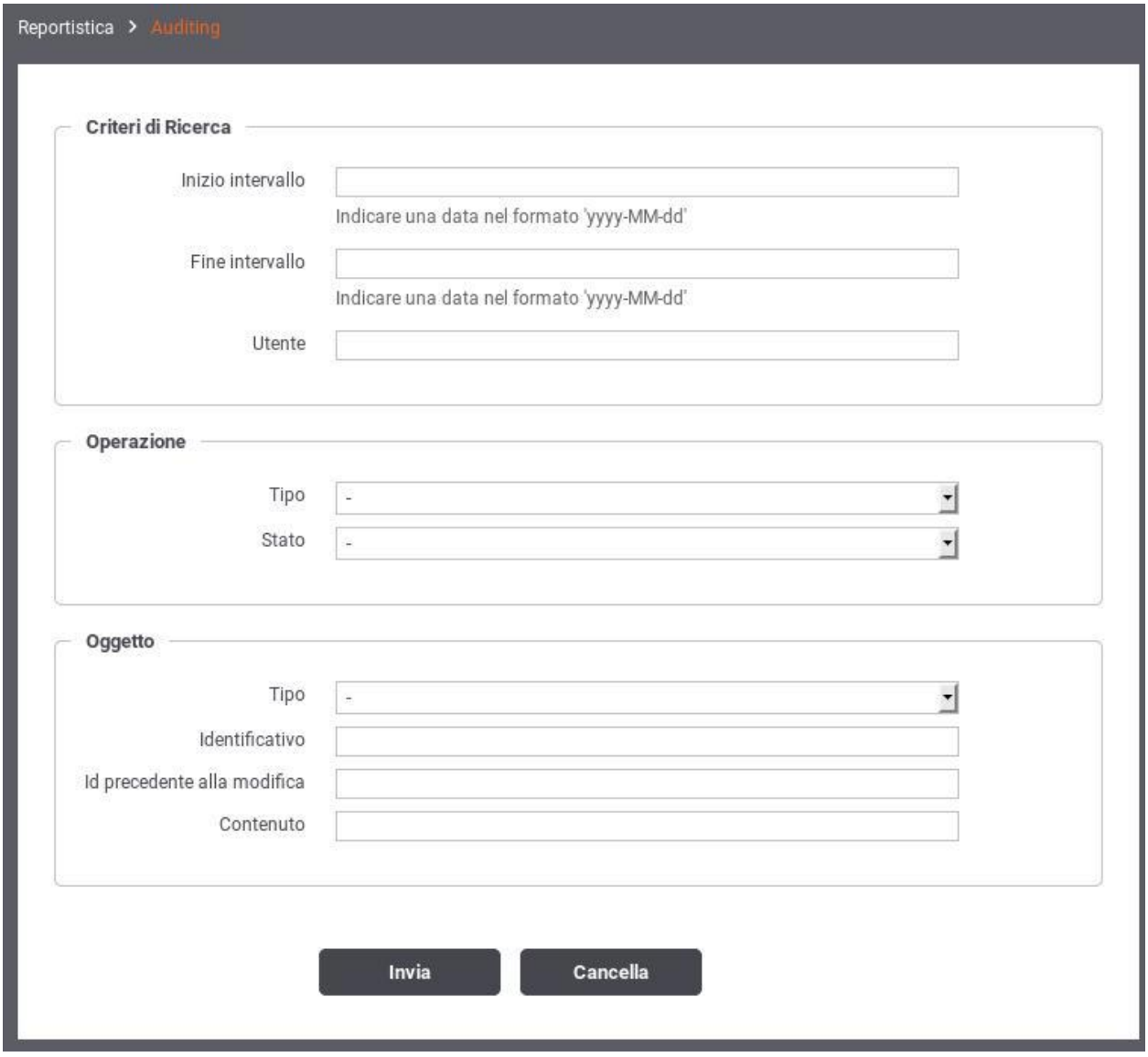
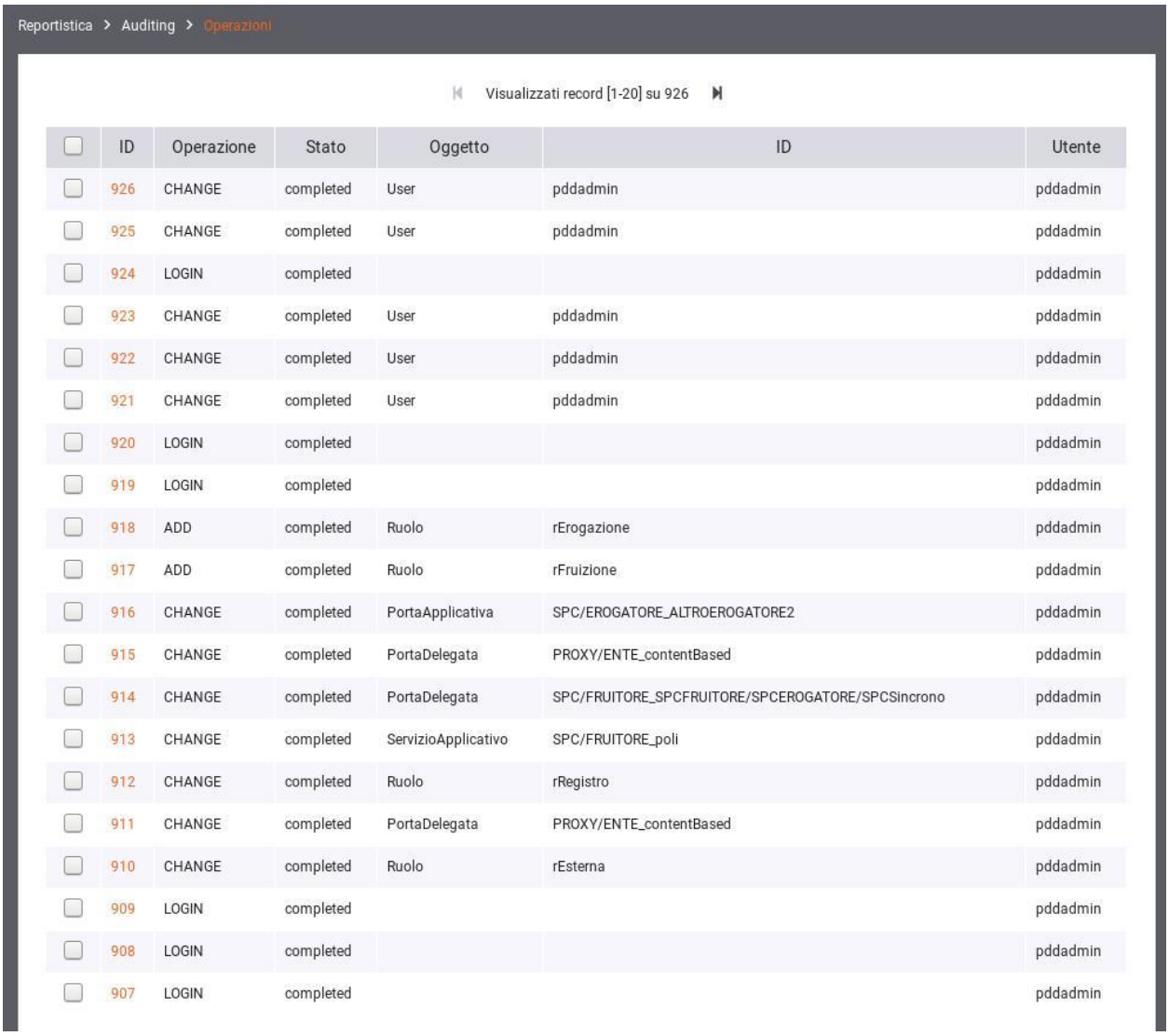


Fig. 7.1: Maschera di ricerca dei dati di auditing

- **Tipo**: campo per restringere la ricerca alle sole operazioni riferite ad un determinato tipo di entità. Il campo è costituito da una lista a discesa popolata con tutte le tipologie di entità gestite dalla govwayConsole.
- **Identificativo**: campo testuale per restringere la ricerca alle sole operazioni effettuate su una specifica entità. La composizione dell'identificativo cambia in base alla tipologia dell'entità. Ad esempio un soggetto è identificato attraverso il tipo e il nome: Tipo/NomeSoggetto.
- **Id precedente alla modifica**: campo testuale analogo al precedente ma utile in quei casi in cui l'operazione che si sta cercando ha modificato i dati che compongono l'identificativo.
- **Contenuto**: pattern per la ricerca sul contenuto dell'entità associata all'operazione. Per utilizzare questo criterio di filtro il servizio di auditing deve essere configurato in modo da effettuare il dump degli oggetti.

Una volta effettuata la ricerca viene mostrata una pagina con la lista dei risultati corrispondenti (vedi Fig. 7.2).



ID	Operazione	Stato	Oggetto	ID	Utente
926	CHANGE	completed	User	pddadmin	pddadmin
925	CHANGE	completed	User	pddadmin	pddadmin
924	LOGIN	completed			pddadmin
923	CHANGE	completed	User	pddadmin	pddadmin
922	CHANGE	completed	User	pddadmin	pddadmin
921	CHANGE	completed	User	pddadmin	pddadmin
920	LOGIN	completed			pddadmin
919	LOGIN	completed			pddadmin
918	ADD	completed	Ruolo	rErogazione	pddadmin
917	ADD	completed	Ruolo	rFruizione	pddadmin
916	CHANGE	completed	PortaApplicativa	SPC/EROGATORE_ALTROEROGATORE2	pddadmin
915	CHANGE	completed	PortaDelegata	PROXY/ENTE_contentBased	pddadmin
914	CHANGE	completed	PortaDelegata	SPC/FRUITORE_SPCFRUITORE/SPCEROGATORE/SPCSincrono	pddadmin
913	CHANGE	completed	ServizioApplicativo	SPC/FRUITORE_poli	pddadmin
912	CHANGE	completed	Ruolo	rRegistro	pddadmin
911	CHANGE	completed	PortaDelegata	PROXY/ENTE_contentBased	pddadmin
910	CHANGE	completed	Ruolo	rEsterna	pddadmin
909	LOGIN	completed			pddadmin
908	LOGIN	completed			pddadmin
907	LOGIN	completed			pddadmin

Fig. 7.2: Risultato della ricerca dei dati di auditing

Ciascun elemento della lista riporta i dati principali che identificano l'operazione. Selezionando l'identificatore dell'operazione si visualizzano i dati di dettaglio (vedi Fig. 7.3). Dal dettaglio dell'operazione, se è attivo il dump, si può

visualizzare il dettaglio dell'entità coinvolta nell'operazione e gli eventuali documenti binari (ad esempio i file WSDL associati ad un accordo di servizio).

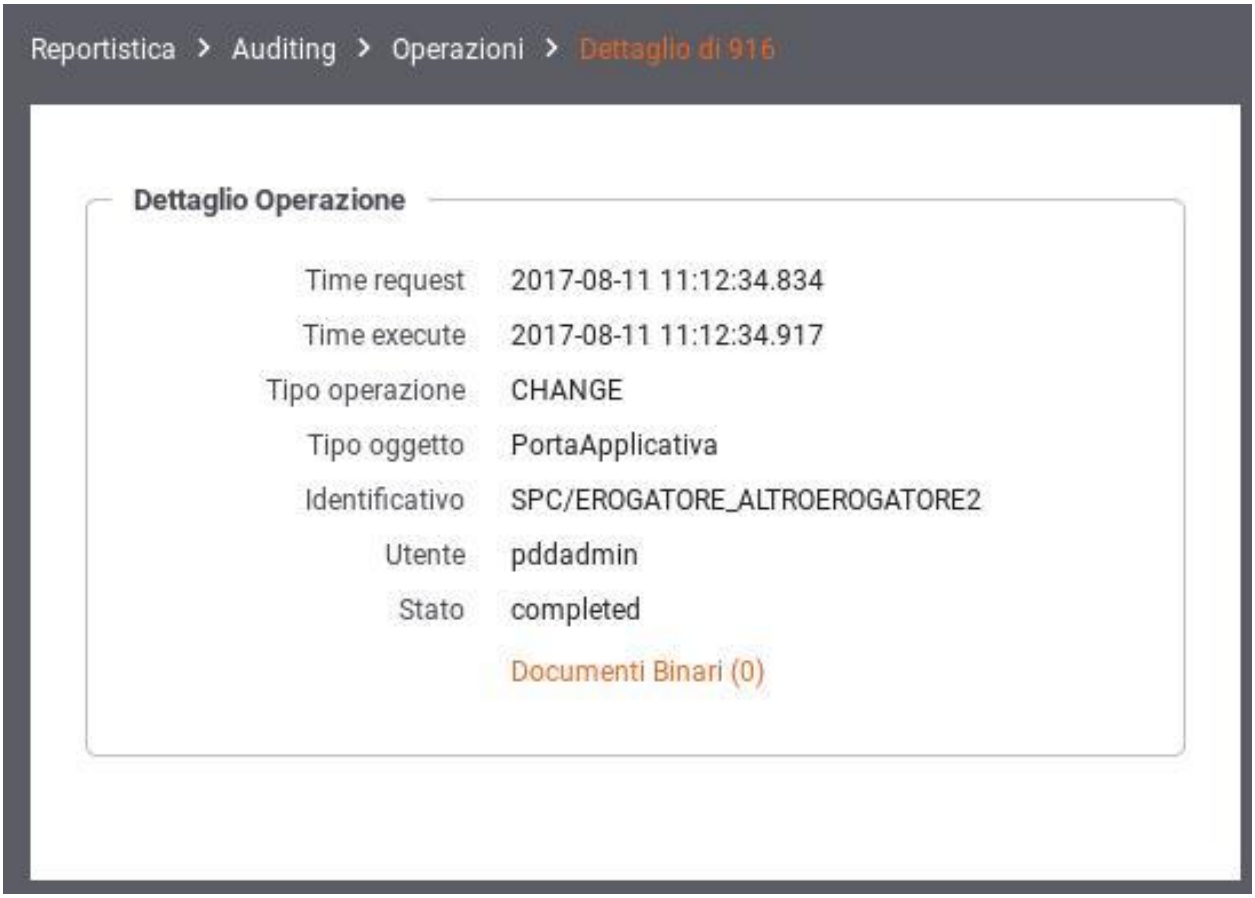


Fig. 7.3: Dettaglio di una traccia di auditing

CAPITOLO 8

Configurazione

Nella sezione del menu *Configurazione* si raggiungono le funzionalità per modificare i parametri di configurazione del gateway.

8.1 Generale

La sezione *Configurazione* > *Generale* consente di impostare i parametri generali per le funzionalità di base del gateway (Fig. 8.1). In particolare è possibile:

- Attivare e configurare la modalità Multi-Tenant. Abilitando questa modalità sarà ammessa la creazione di ulteriori soggetti interni al dominio GovWay.
- Configurare le Base URL utilizzate per visulizzare le URL di invocazione delle API
- Configurare la gestione del CORS (*cross-origin HTTP request (CORS)*) a livello globale valido per tutte le APIs
- Configurare il Caching Risposta a livello globale valido per tutte le APIs
- Configurare i profili fornendo i riferimenti ai servizi di base per l'elaborazione dei messaggi ed al soggetto interno
- Attivare e configurare la modalità Canali in una installazione composta da più nodi in Load Balancing. Abilitando questa modalità sarà possibile assegnare uno o più canali ad ogni nodo che compone il cluster e suddividere le API in canali di appartenenza. Su ogni nodo saranno autorizzate ad essere invocate solamente le API che possiedono un canale corrispondente alla configurazione del nodo.
- Configurare proprietà di sistema

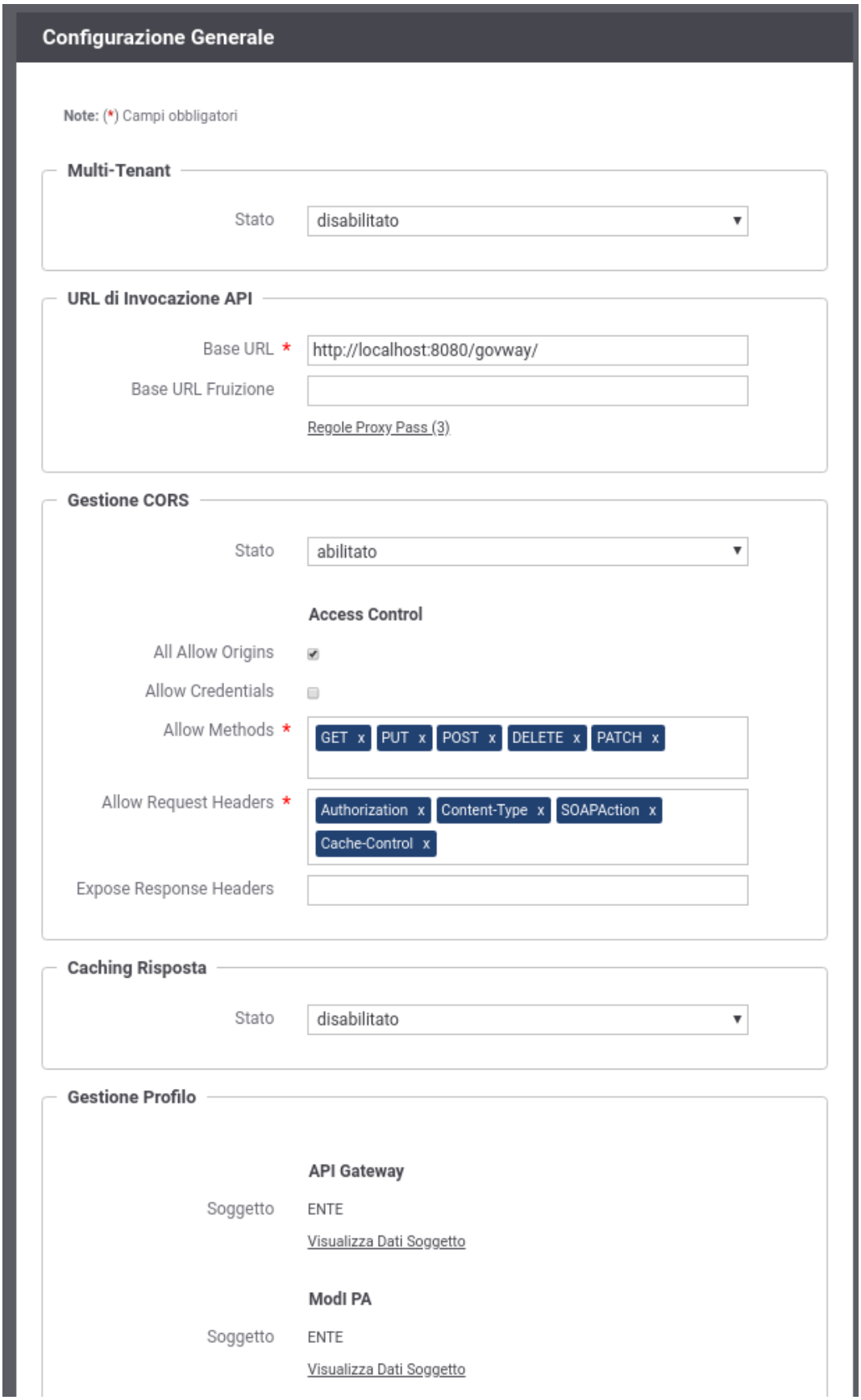


Fig. 8.1: Maschera per l'impostazione dei parametri di configurazione generale (1/2)

Fig. 8.2: Maschera per l'impostazione dei parametri di configurazione generale (2/2)

8.1.1 Multi-Tenant

Per abilitare la modalità multi-tenant è sufficiente selezionare il valore «abilitato» sull'elemento Stato.

Dopo aver abilitato l'opzione multi-tenant è possibile creare nuovi soggetti interni al dominio, come indicato alla sezione *Creazione di un soggetto*. In questo contesto, i soggetti avranno come elemento distintivo il dominio, che può essere *Interno* o *Esterno*.

I dettagli sulla configurazione dell'opzione multi-tenant sono riportati nella sezione *Multi-Tenant*.

8.1.2 URL di Invocazione API

Questa sezione visualizza:

- *Base URL*: Indica il prefisso utilizzato per visualizzare le URL di Invocazione delle API.
- *Base URL Fruizione*: permette di differenziare il prefisso utilizzato per visualizzare le URL di Invocazione delle fruizioni dalle erogazioni.
- *Regole Proxy Pass*: tramite questa voce è possibile ridefinire le URL di Invocazioni, per specifiche fruizioni e/o erogazioni, allineandole a regole configurate su un revere proxy che media le comunicazioni http con GovWay.

Regole Proxy Pass

Questa sezione permette di ridefinire la modalità di visualizzazione delle Url di Invocazione delle API esposte da GovWay per assicurare che, in presenza di un reverse proxy che media le comunicazioni http con GovWay, sia possibile configurare opportunamente le url di invocazione delle API esposte da GovWay allineandole con le eventuali configurazioni specifiche realizzate sul reverse proxy.

> **Nota:** La funzionalità permette di configurare come vengono visualizzate le URL di Invocazione sulla govwayConsole, per allinearsi ad un eventuale reverse proxy che media le comunicazioni http con GovWay. Le API, su GovWay, rimangono raggiungibili solamente sulle url originali e dovrà essere il reverse proxy ad effettuare la conversione rispetto a quella esposta.

Le regole create sono visualizzate nella forma di elenco ordinato (Fig. 8.3). L'icona iniziale di ciascun elemento consente di modificarne la posizione. Per ogni regola viene visualizzato il suo stato, il nome e la descrizione.

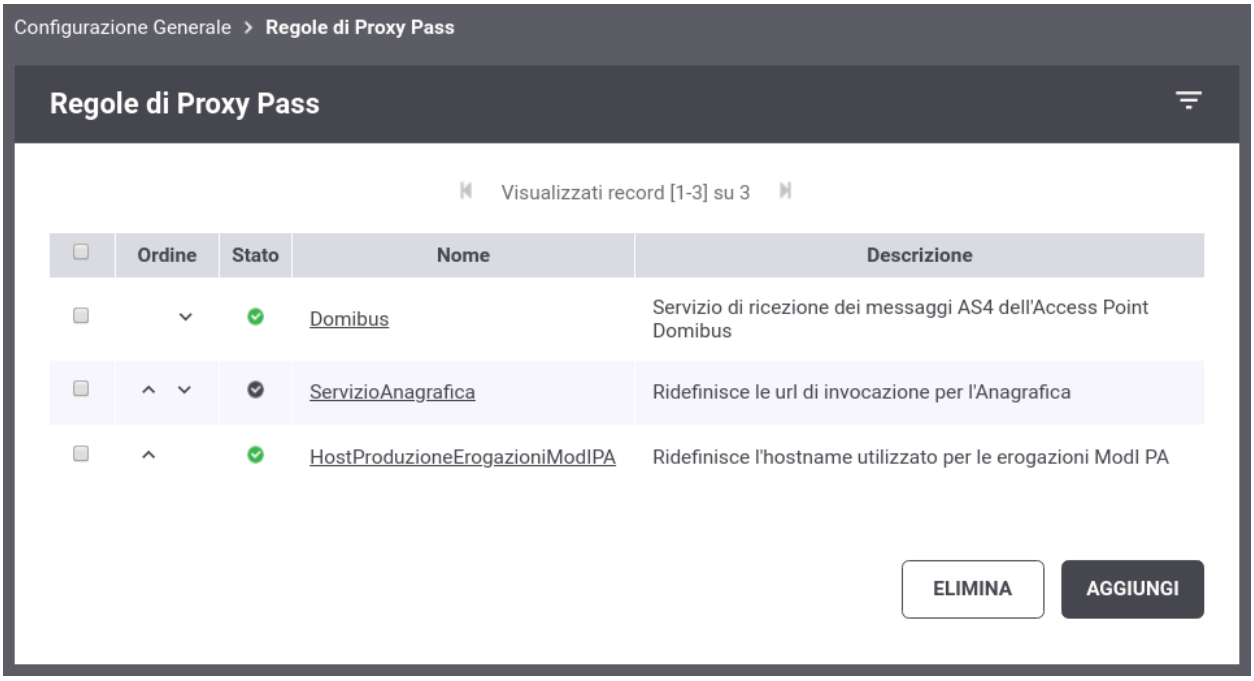


Fig. 8.3: Lista Regole Proxy Pass

Per ogni regola (Fig. 8.4) deve essere obbligatoriamente definita una stringa libera o una espressione regolare utilizzata per individuare l'applicabilità della regola attraverso un confronto con il contesto dell'API. Il contesto è l'URL di Invocazione dell'API senza il prefisso Base URL. Inoltre per ogni regola è possibile indicare altri criteri di applicabilità opzionali quali eventuali profilo di interoperabilità, un soggetto, una tipologia (fruizione/erogazione) o un tipo di api (soap/rest).

Il dettaglio dei campi associati ad una regola sono raggruppati in tre sottosezioni:

Informazioni generiche:

- *Nome*: Identificativo della regola
- *Stato*: Indica se la regola è abilitata o meno.
- *Descrizione*: (Opzionale) Descrizione generica della regola

Le regole di applicabilità vengono definite dai seguenti campi:

- *Espressione Regolare*: Indica se la regola sottostante è una espressione regolare o una stringa libera.
- *Regola*: Stringa libera o espressione regolare.
  - L'espressione regolare viene utilizzata per verificarne il match sull contesto dell'API (url di invocazione senza la Base URL)
  - Nel caso di stringa libera si ha un'applicabilità se il contesto dell'API (url di invocazione senza la Base URL) inizia con la stringa fornita.
- *Profilo*: (Opzionale) Profilo di Interoperabilità per il quale si applica la regola
- *Soggetto*: (Opzionale) Soggetto interno per il quale si applica la regola
- *Ruolo*: (Opzionale) Tipologia di API (Erogazione/Fruizione) per il quale si applica la regola
- *Tipo API*: (Opzionale) Tipo di API (REST/SOAP) per il quale si applica la regola

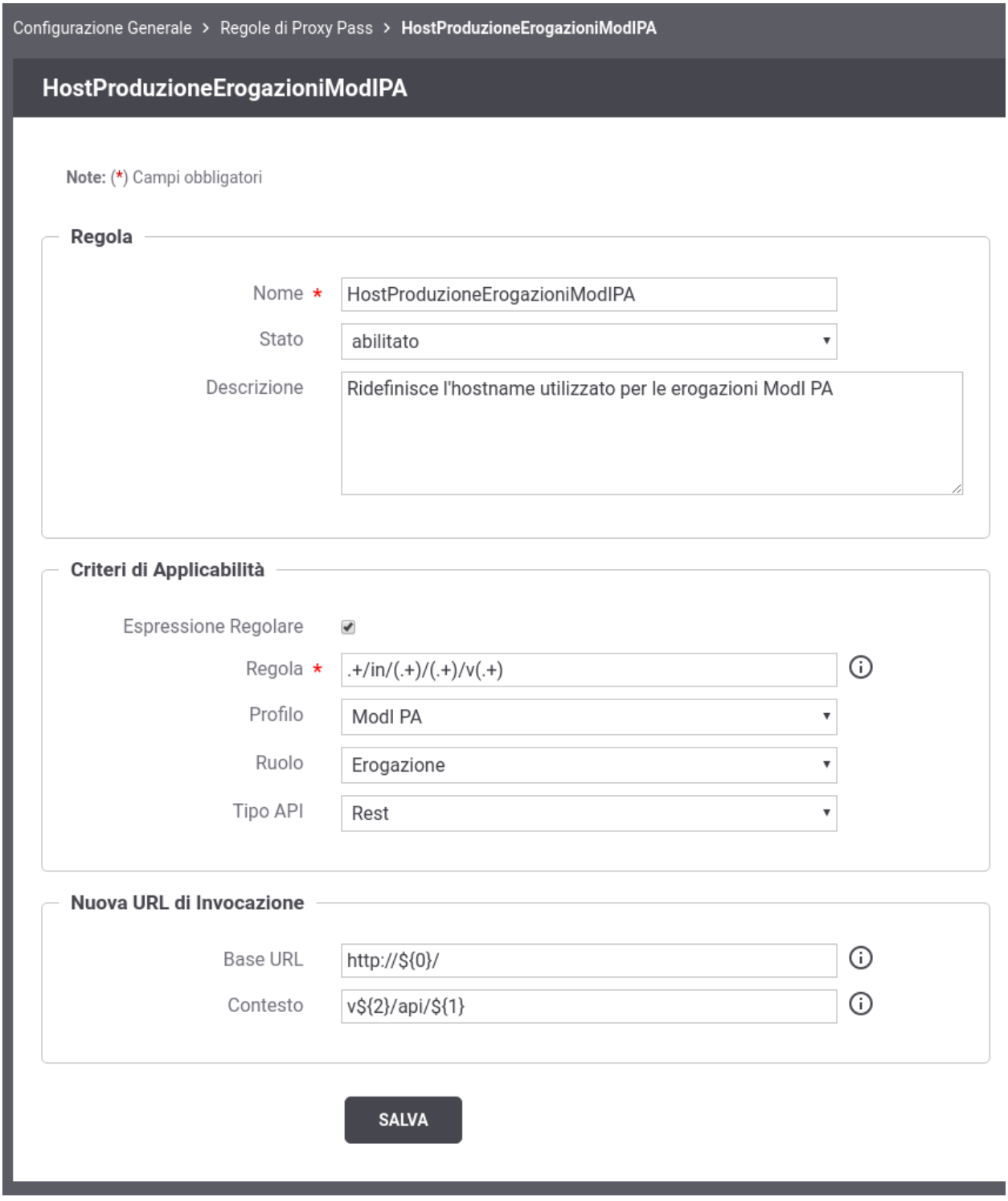


Fig. 8.4: Creazione Regola Proxy Pass

La nuova url di invocazione viene definita attraverso i campi "Base URL" e "Contesto".

- *Base URL*: Permette di ridefinire la Base URL utilizzata rispetto a quanto definito nella configurazione generale
- *Contesto*: Indica il contesto da utilizzare dopo la Base URL

Nei campi "Base URL" e "Contesto" è possibile utilizzare le seguenti informazioni dinamiche:

- Se è stata fornita una espressione regolare, nei due campi possono essere utilizzati le keyword "${posizione}" per impostare un valore dinamico individuato tramite l'espressione regolare fornita. Il primo match, all'interno dell'espressione regolare, è rappresentata da "${0}" (Ad esempio: http://server:8080/${0}/altro/${1}/)
- Se è abilitata la gestione dei canali (vedi *Canali*) è possibile utilizzare la keyword "${canale}" per impostare l'identificativo del canale associato all'API. Maggiori esempi vengono forniti nella sezione *Url di Invocazione e Canali*.
- È possibile utilizzare la keyword "${tag}" per impostare l'identificativo del tag associato all'API. Poichè ad un'API è possibile associare più tag verrà utilizzato quello alla prima posizione ma è possibile indicarne uno differente tramite l'espressione ${tag[posizione]}. Il primo tag, all'interno della lista, è rappresentata da "${tag[0]}", ad esempio: http://server:8080/${tag[0]}/

Esempio 1

Tutte le API REST erogate dal Soggetto "ENTE" tramite il profilo "ModI" possiedono nell'installazione di default la seguente URL di Invocazione:

- http://localhost:8080/rest/in/ENTE/NomeAPI/v1

Per modificare la url di invocazione in modo da spostare il nome del soggetto come hostname, e rimodulare il contesto in modo da visualizzare prima la versione, è possibile utilizzare la seguente configurazione di proxy pass:

Criteri di Applicabilità:

- Espressione Regolare: true
- Regola: .+/in/(.+)/(.+)/v(.+)
- Profilo: ModI
- Soggetto: ENTE
- Ruolo: Erogazione
- Tipo API: REST

Nuova URL di Invocazione

- Base URL: http://${0}/
- Contesto: v${2}/api/${1}

L'url di invocazione prodotta sarà:

- http://ENTE/v1/api/NomeAPI

Esempio 2

Supponiamo di voler modificare l'url di invocazione dell'API "PetStore" versione 1 erogata dal soggetto "ENTE" tramite il profilo di interoperabilità "ModI". Nell'installazione di default viene fornita la seguente URL di Invocazione:

- http://localhost:8080/rest/in/ENTE/PetStore/v1

Lo scopo è quello di eliminare il nome del soggetto e di togliere la "v" dalla versione. Per farlo è possibile utilizzare la seguente configurazione di proxy pass:

Criteri di Applicabilità:

- Espressione Regolare: false

- Regola: /rest/in/ENTE/PetStore/v1
- Profilo: ModI
- Soggetto: Qualsiasi
- Ruolo: Qualsiasi
- Tipo API: Qualsiasi

Nuova URL di Invocazione

- Base URL:
- Contesto: /rest/in/PetStore/1

L'url di invocazione prodotta sarà:

- http://localhost:8080/rest/in/PetStore/1

8.1.3 Gestione CORS

In GovWay è possibile abilitare la gestione del CORS (*cross-origin HTTP request (CORS)*) globalmente in modo che sia valido per tutte le APIs.

La configurazione permette di specificare i seguenti parametri:

- *Stato*: Indicazione se la gestione del CORS è abilitata o meno globalmente su GovWay.
- *Access Control*: tutti i parametri seguenti permettono di configurare il CORS. Per il dettaglio su cosa significa ogni singola voce si rimanda alla specifica CORS *https://www.w3.org/TR/cors/*.
  - *All Allow Origins*: se abilitato, in tutte le risposte viene aggiunto un header http "Access-Control-Allow-Origin" valorizzato con "*"
  - *Allow Origins*: nel caso non venga abilitato il parametro precedente, deve essere indicato una lista di origin che vengono impostate nell'header http "Access-Control-Allow-Origin" aggiunto in ogni risposta
  - *All Allow Methods*: se abilitato, in tutte le risposte di una Preflight Request (OPTIONS) viene aggiunto un header http "Access-Control-Allow-Methods" valorizzato con i metodi richiesti dall'header "Access-Control-Request-Method" della richiesta. L'opzione è attivabile solamente se la voce "All Allow Origins" risulta disabilitata
  - *Allow Methods*: nel caso non venga abilitato il parametro precedente, deve essere indicato una lista di metodi che vengono impostati nell'header http "Access-Control-Allow-Methods" di una risposta Preflight Request (OPTIONS)
  - *All Allow Request Headers*: se abilitato, in tutte le risposte di una Preflight Request (OPTIONS) viene aggiunto un header http "Access-Control-Allow-Headers" valorizzato con gli header http richiesti dall'-header "Access-Control-Request-Headers" della richiesta. L'opzione è attivabile solamente se la voce "All Allow Origins" risulta disabilitata
  - *Allow Request Headers*: nel caso non venga abilitato il parametro precedente, deve essere indicato una lista di header che vengono impostati nell'header http "Access-Control-Allow-Headers" di una risposta Preflight Request (OPTIONS)
  - *Allow Credentials*: se abilitato o disabilitato viene impostato relativemente il valore true o false nell'header "Access-Control-Allow-Credentials"
  - *Expose Response Headers*: abilita l'accesso a specifici headers, presenti nella risposta, da parte dei client.

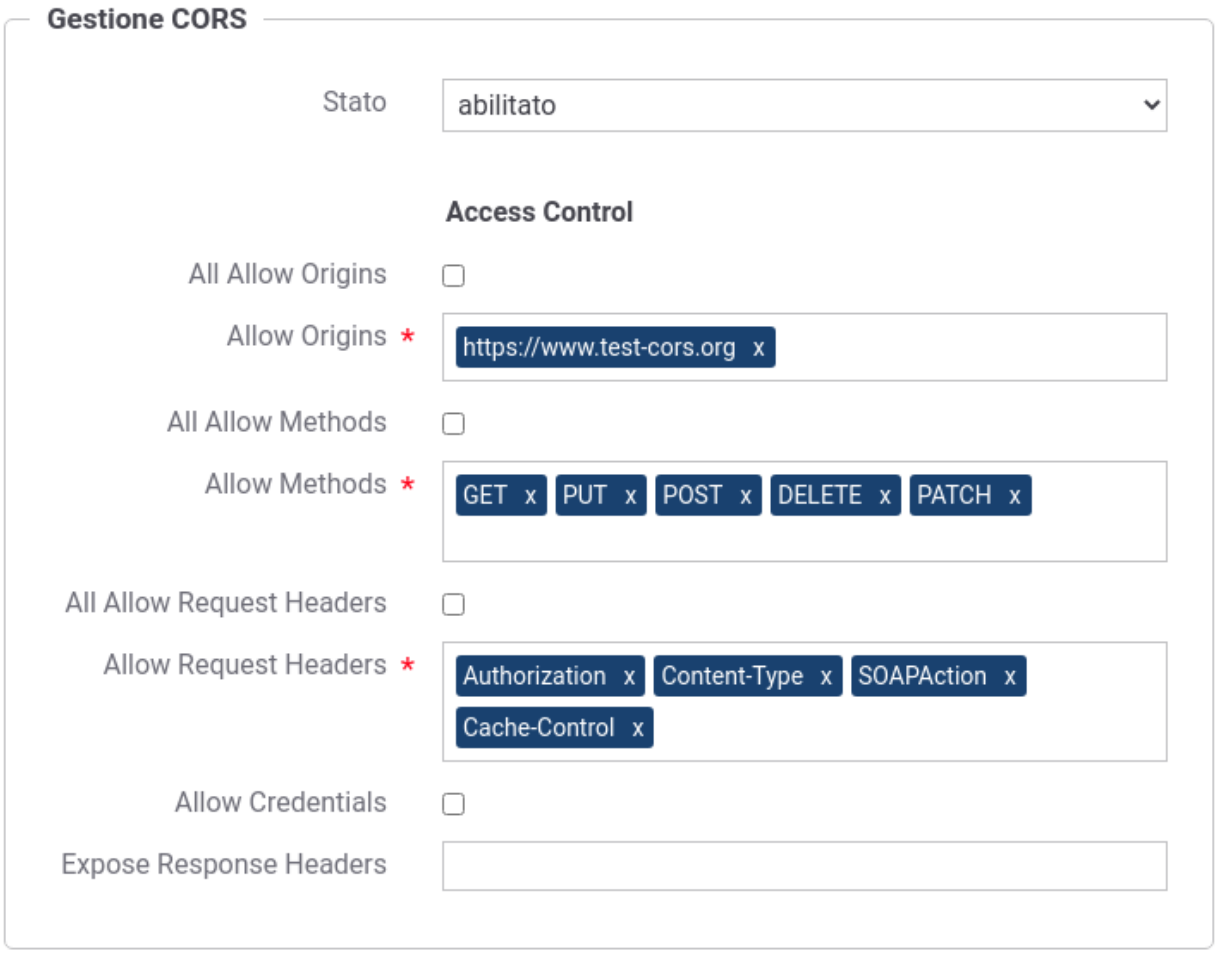


Fig. 8.5: Maschera di configurazione generale del CORS

8.1.4 Caching Risposta

In GovWay è possibile abilitare il salvataggio delle risposte in una cache globalmente in modo che sia attivo per tutte le APIs. Questa funzionalità permette ad un backend server di non dover riprocessare le stesse richieste più volte.

La configurazione permette di specificare i seguenti parametri:

- *Stato*: Indicazione se il salvataggio delle risposte in cache è abilitata o meno globalmente su GovWay.
- *Cache Timeout (secondi)*: intervallo di tempo, definito in secondi, per il quale la risposta salvata in cache viene utilizzata come risposte a successive richieste di un client.
- *Dimensione Max Risposta*: se abilitato deve essere definita la dimensione massima (in kb) che una risposta può avere per essere salvata in cache.
- *Generazione Hash*: ad ogni risposta salvata in cache viene associato un valore hash calcolato rispetto ai dati della richiesta che risultano abilitati tra le opzioni seguenti:
  - *URL di Richiesta*: viene utilizzata la URL della richiesta per il calcolo dell'hash.
  - *Payload*: viene utilizzato il payload della richiesta per il calcolo dell'hash.
  - *Headers*: vengono utilizzati gli header della richiesta indicati per il calcolo dell'hash. L'abilitazione di questa opzione comporta l'aggiunta di un elemento per consentire di specificare gli headers da selezionare.
- *Cache Control*: opzioni aggiuntive per la gestione della cache basate sul header HTTP «Cache-Control»:
  - *No Cache*: consente di attivare l'utilizzo della direttiva «no-cache» al fine di effettuare una richiesta evitando di ottenere una risposta dalla cache.
  - *Max Age*: consente di attivare l'utilizzo della direttiva «max-age» che consente di forzare il tempo di vita, al valore fornito, della risposta inserita in cache.
  - *No Store*: consente di attivare l'utilizzo della direttiva «no-store» che consente di impedire l'inserimento in cache della risposta generata dalla richiesta corrente.

Dopo aver salvato la configurazione fornita per il caching della risposta, appare la sezione *Configurazione Avanzata* che comprende il link *Regole*. Seguendo tale link è possibile definire ulteriori criteri avanzati per la gestione della cache.

Nota: In presenza di regole avanzate di configurazione, le risposte salvate in cache saranno solamente quelle che hanno un match con i criteri definiti in una regola.

Come si vede in Fig. 8.7 ciascuna regola è composta dai seguenti campi:

- *Codice Risposta*: codice HTTP ottenuto in risposta. Sono disponibili per la scelta le seguenti opzioni:
  - *Qualsiasi*: indica qualunque valore del codice HTTP restituito
  - *Singolo*: consente di specificare un singolo valore del codice HTTP restituito
  - *Intervallo*: consente di fornire l'intervallo dei valori ammessi per il codice HTTP restituito
- *Cache Timeout (Secondi)*: indica in secondi il timeout applicato agli elementi in cache relativamente ai codici HTTP che soddisfano la regola.
- *Fault*: opzione per specificare se anche i messaggi di fault devono essere inseriti in cache.

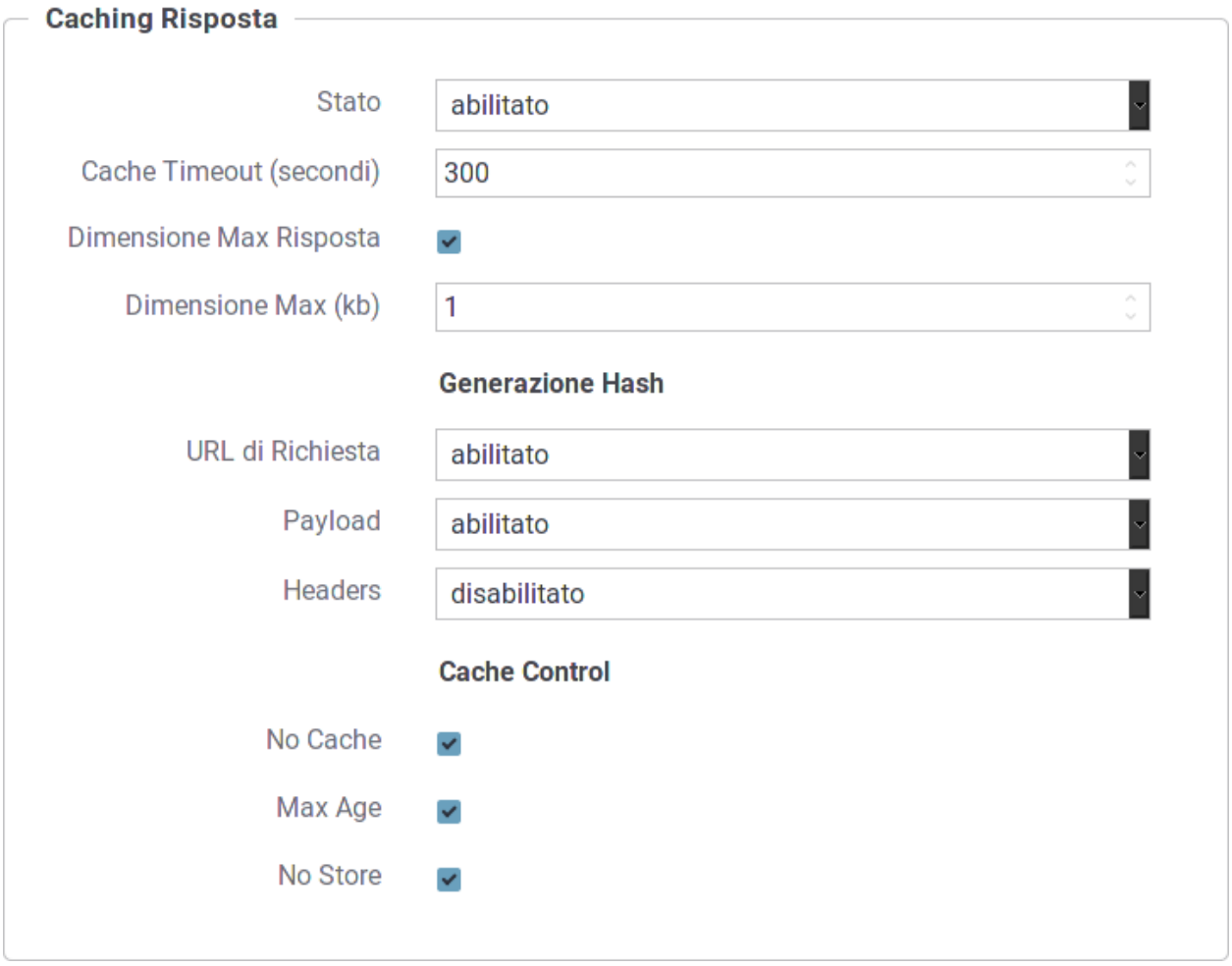


Fig. 8.6: Maschera di configurazione per il Caching della Risposta

Fig. 8.7: Inserimento di una regola per il Caching della Risposta

8.1.5 Profili

Questa sezione viene visualizzata solamente se non è attiva la modalità Multi-tenant. Per ciascun Profilo di Interoperabilità, attivo sul gateway, viene visualizzato il nome del Soggetto interno che eroga/fruisce. Subito sotto il soggetto è presente un collegamento che porta al form di editing del soggetto visualizzato.

8.1.6 Canali

In GovWay è possibile attivare, in una installazione composta da più nodi in Load Balancing, una suddivisione delle API tra i vari nodi utilizzando il concetto di canale, al fine di suddividere il carico tra i nodi. Per maggiori dettagli sull'installazione in Load Balancing si faccia riferimento alla sezione cluster della Guida di Installazione.

Abilitando la modalità "Canali" sarà possibile assegnare uno o più canali ad ogni nodo che compone il cluster ed un canale ad ogni API. Su ogni nodo saranno autorizzate ad essere invocate solamente le API che possiedono un canale corrispondente alla configurazione del nodo.

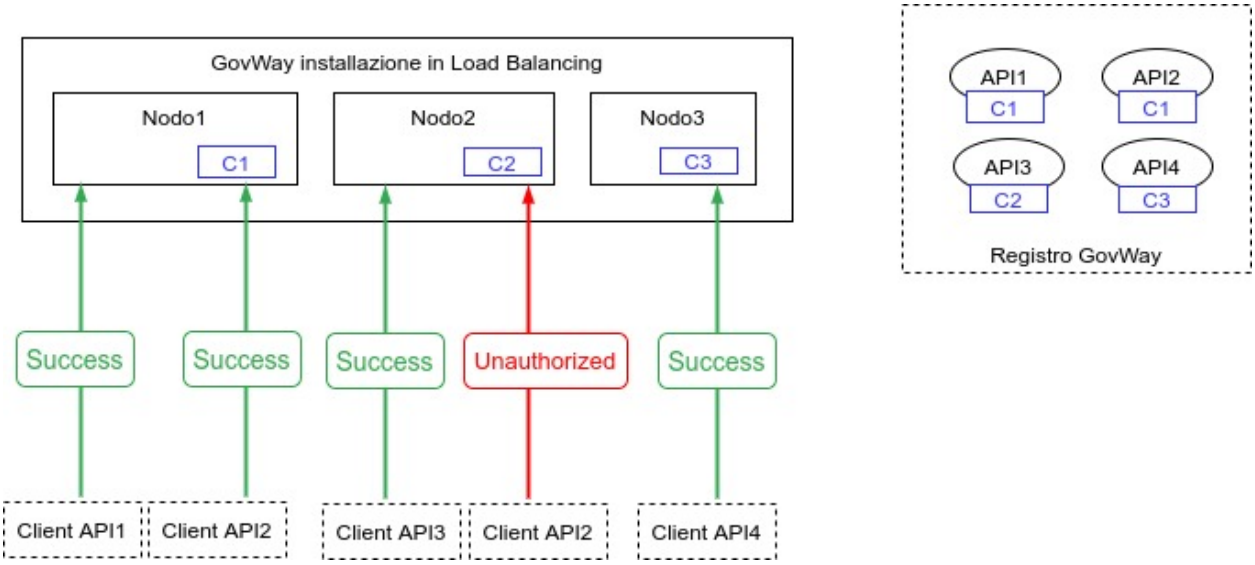


Fig. 8.8: Suddivisione delle API in Canali

Nelle prossime sezioni verranno descritte:

- *Configurazione dei Canali*: vengono fornite le indicazioni su come abilitare la funzionalità e su come censire i canali ed associarli ai nodi che compongono il cluster.
- *Canale associato all'API*: vengono fornite le indicazioni su come associare ad una erogazione o fruizione di API un canale differente da quello di default.
- *Url di Invocazione e Canali*: viene descritto come personalizzare le url di invocazione visualizzate dalla console per ogni erogazione o fruizione al fine di indirizzare il nodo corretto corrispondente al canale associato all'API.

Configurazione dei Canali

Abilitando la modalità "Canali" sarà possibile assegnare uno o più canali ad ogni nodo che compone il cluster ed un canale ad ogni API.

La prima volta che viene abilitata la funzionalità, la console richiede di configurare un canale di default che verrà associato:

- a tutte le erogazioni o fruizioni di API esistenti alle quali non è stato associato un canale
- a tutti i nodi non registrati

La configurazione richiede (Fig. 8.9):

- *Stato*: indicazione se la gestione è abilitata o meno;
- *Nome*: identificativo univoco del canale di default;
- *Descrizione*: descrizione generica del canale di default.

Fig. 8.9: Maschera di abilitazione delle gestione dei canali

Attivata la gestione e definito il canale di default sarà possibile registrare nuovi canali, modificare il canale di default e registrare i nodi che compongono il cluster (Fig. 8.10).

Fig. 8.10: Maschera di gestione dei canali

Dall'elenco dei canali è possibile aggiungere, modificare o eliminare un canale (Fig. 8.11). La registrazione di un nuovo canale richiede che venga definito un identificativo univoco e opzionalmente una descrizione da associare al canale.

Fig. 8.11: Elenco dei canali configurati

Dall'elenco dei nodi è possibile registrare, modificare o eliminare un nodo del cluster (Fig. 8.11).

Fig. 8.12: Elenco dei nodi configurati

La registrazione o la modifica di un nodo richiede (Fig. 8.13):

- *Nome*: identificativo univoco del nodo;
- *Descrizione*: descrizione generica del nodo;
- *Canali*: selezione dei canali associati al nodo.

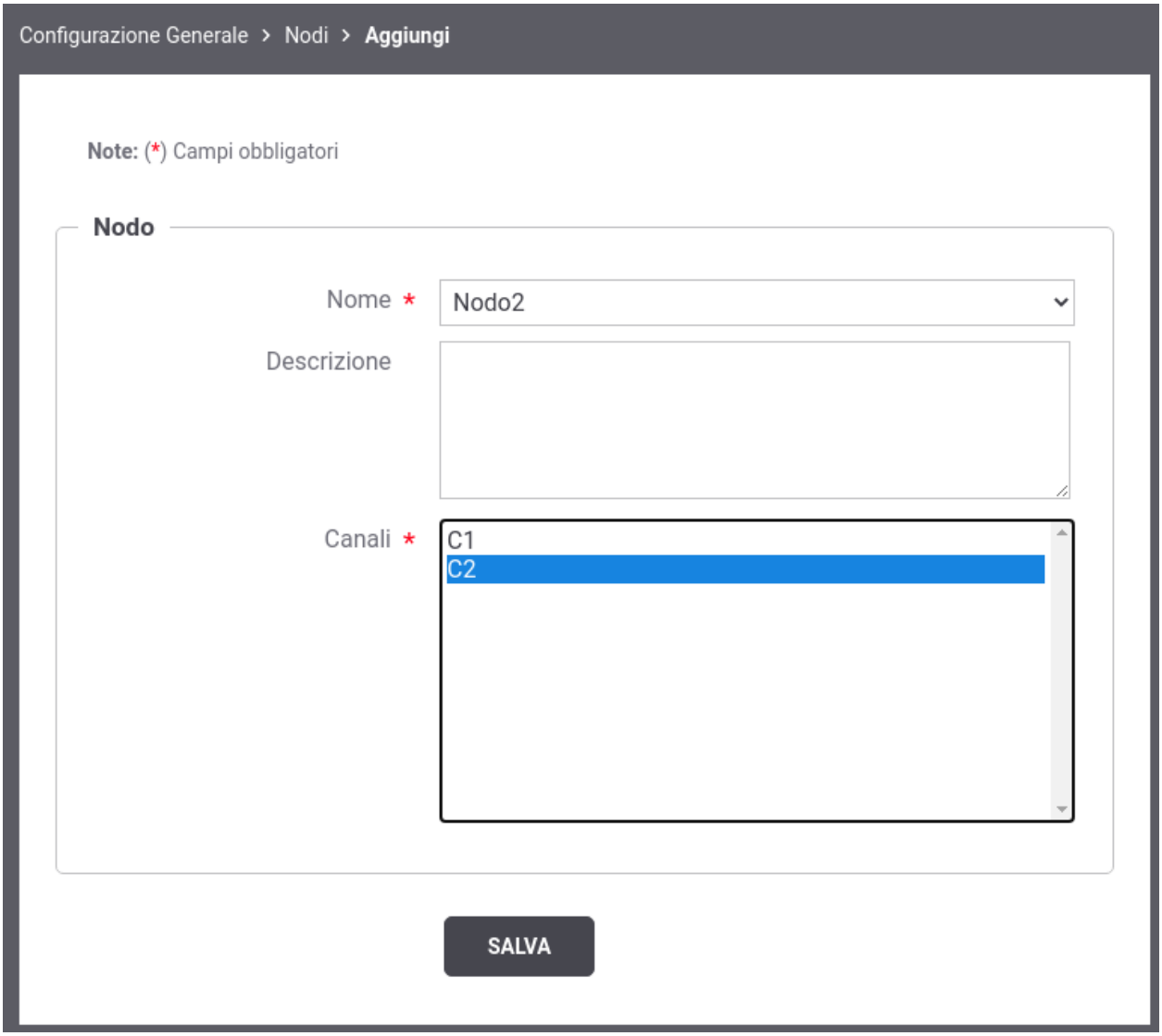


Fig. 8.13: Registrazione di un nodo

Canale associato all'API

Una volta abilitata la modalità "Canali" accedendo all'elenco delle API (Fig. 8.14) o delle Erogazioni/Fruizioni (Fig. 8.15) verrà visualizzata l'informazione sul canale associato. Tutte le erogazioni/fruizioni esistenti in cui non è stato associato un canale specifico ereditano il canale di default.

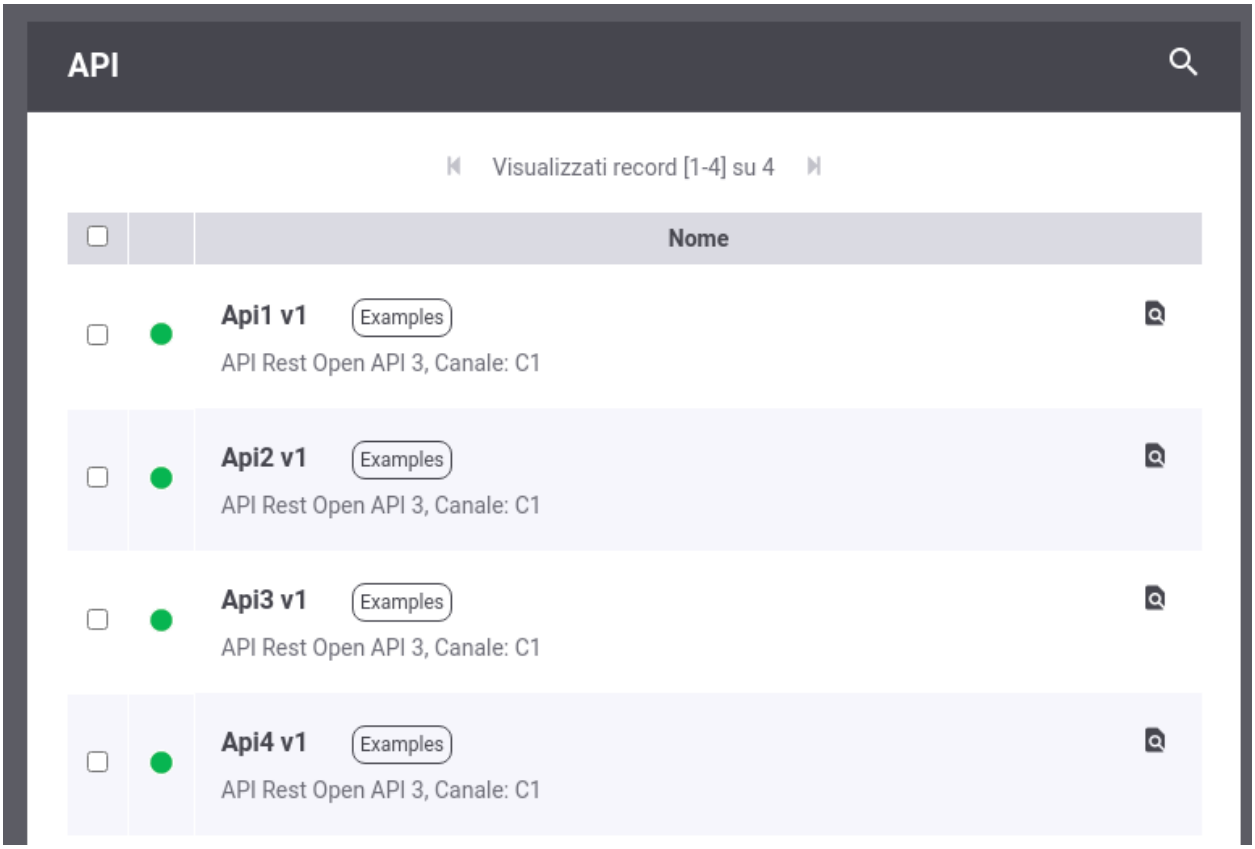


Fig. 8.14: Elenco API visualizza l'informazione sul Canale

Un canale, differente da quello di default, può essere associato ad una erogazione o fruizione in due modi:

- definendo un canale nel'API: tutte le erogazioni o fruizioni che implementano l'API ereditano il canale
- associando il canale alla specifica erogazione o fruizione

Sia durante la registrazione di una nuova API che durante l'attivazione di una nuova erogazione o fruizione verrà richiesto all'utente se desidera specificare un canale differente da quello di default (Fig. 8.16).

È possibile modificare il canale associato ad una API esistente accedendo alla maschera di dettaglio dell'API (Fig. 8.17) e cliccando sulla voce "Modifica Canale" si accede ad una maschera identica a quella proposta in fase di creazione (vedi Fig. 8.16).

In ugual modo è possibile associare un canale ad una specifica erogazione o fruizione accedendo alla sua maschera di dettaglio (Fig. 8.18) e cliccando sulla voce "Modifica Canale".

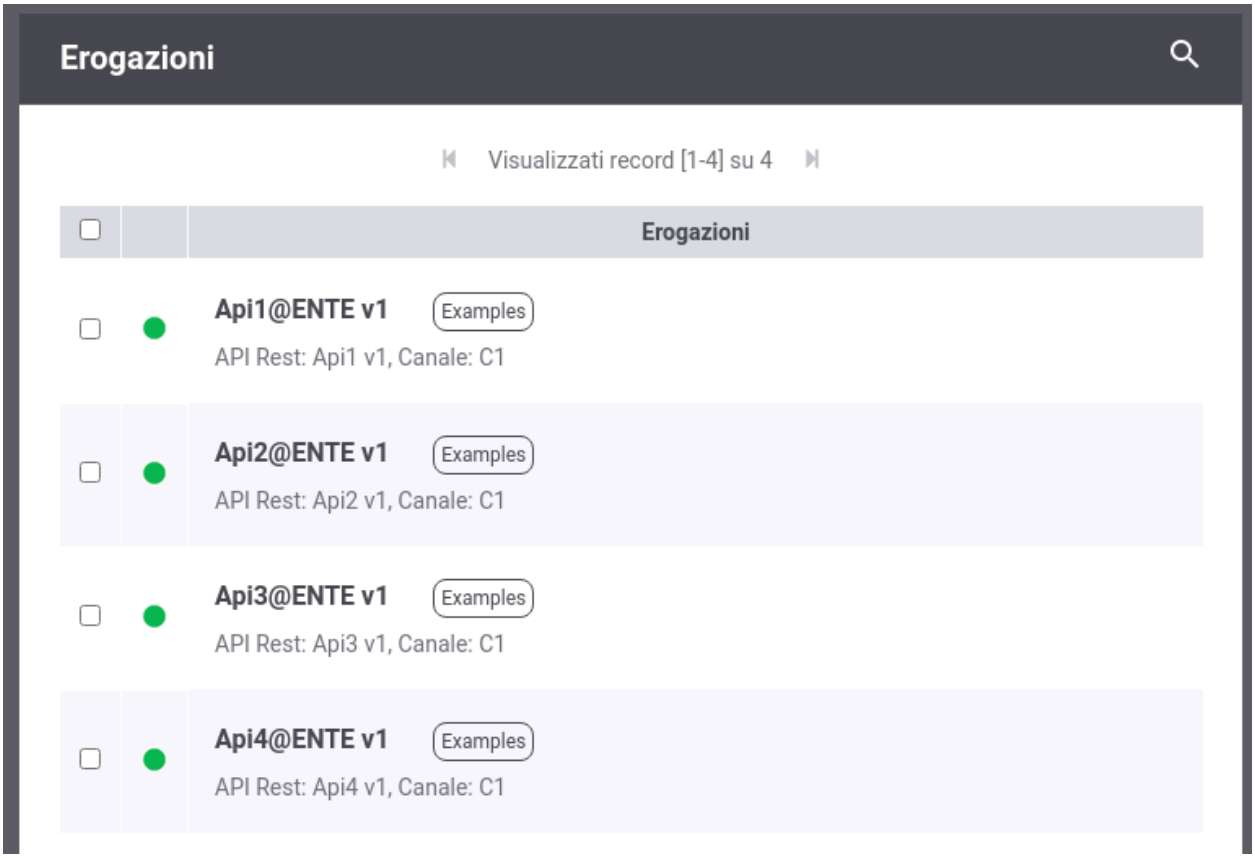


Fig. 8.15: Elenco Erogazioni visualizza l'informazione sul Canale

Fig. 8.16: Associazione di un canale

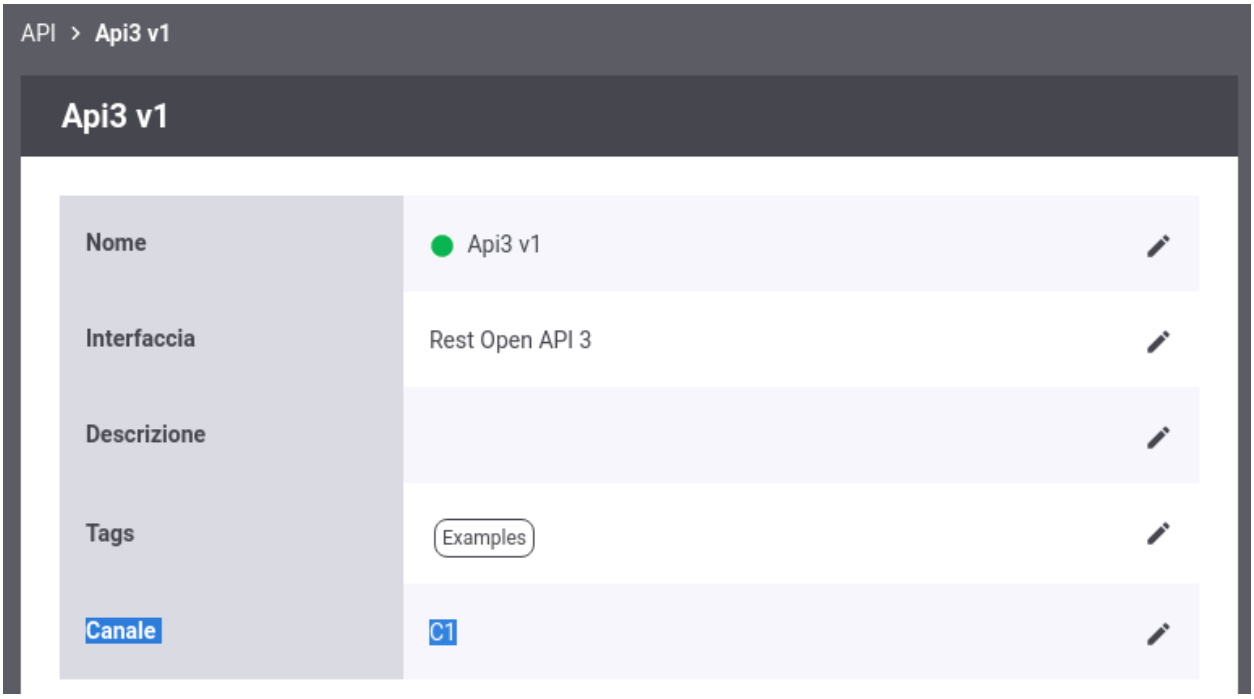


Fig. 8.17: Modifica del canale associato ad un'API

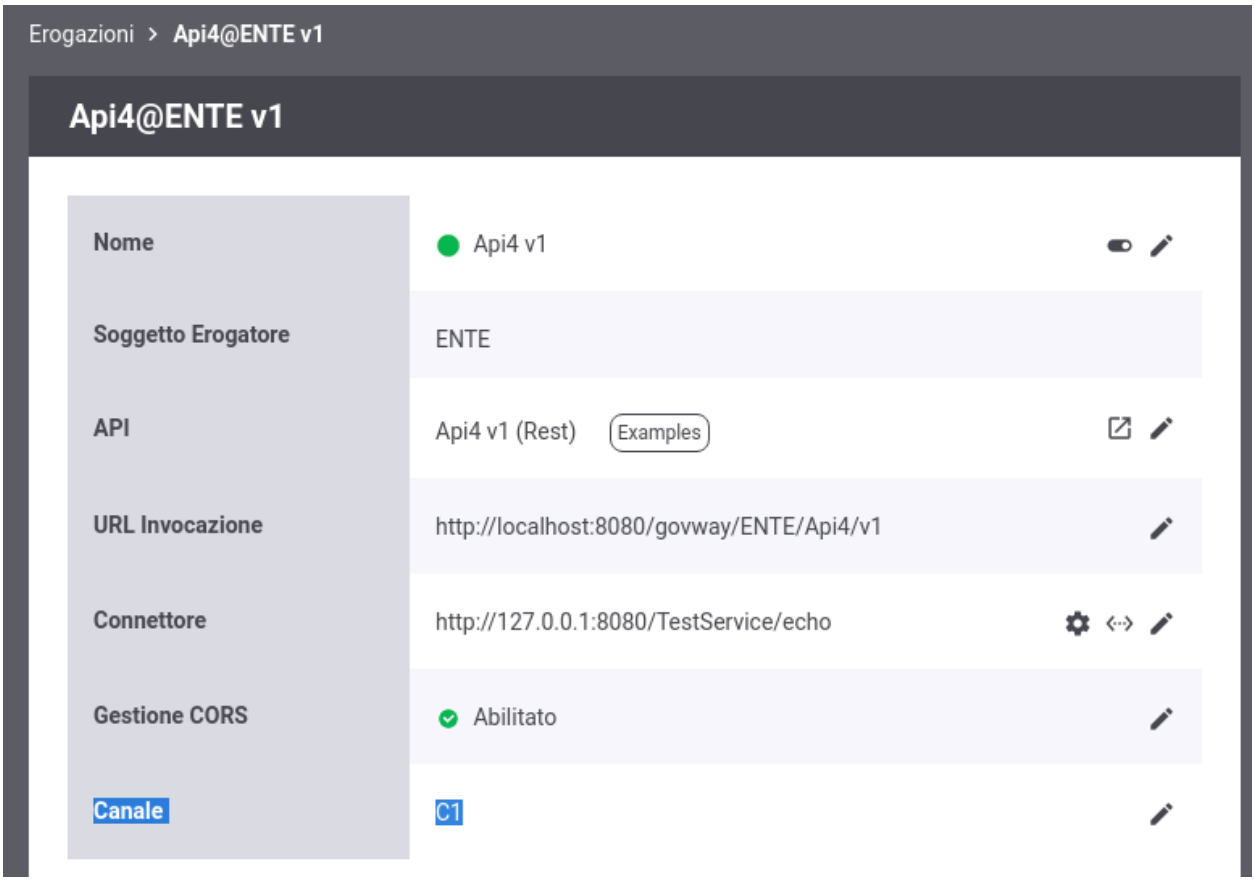


Fig. 8.18: Modifica del canale associato ad una erogazione

Url di Invocazione e Canali

Su ogni nodo saranno autorizzate ad essere invocate solamente le API che possiedono un canale corrispondente alla configurazione del nodo come raffigurato in Fig. 8.19.

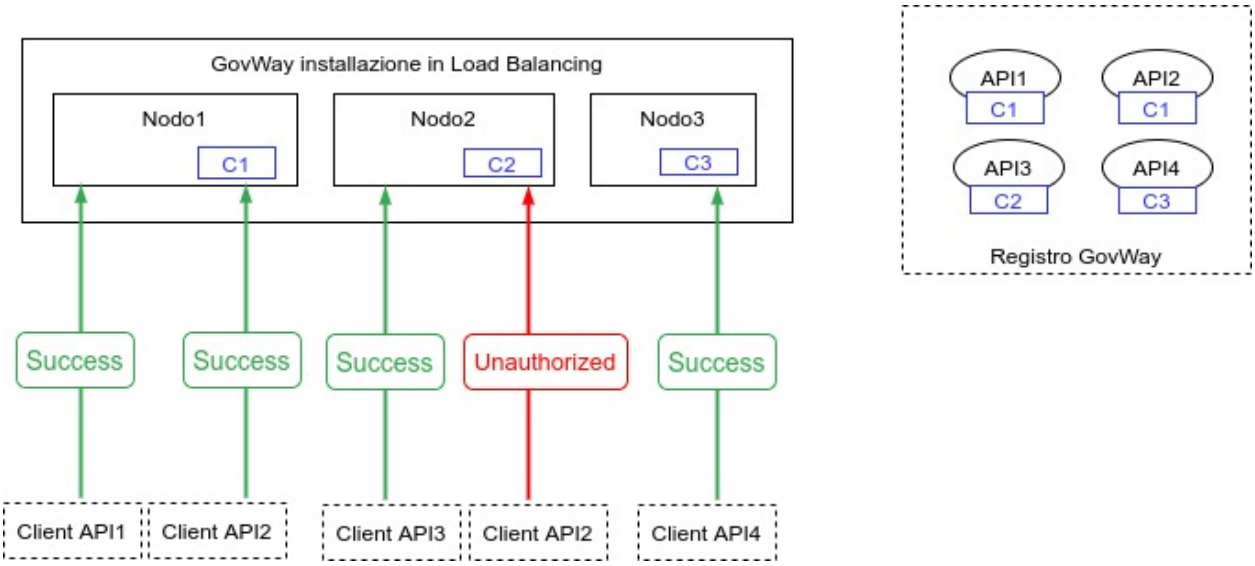


Fig. 8.19: Suddivisione delle API in Canali

È possibile utilizzare le regole di proxy pass descritte nella sezione *URL di Invocazione API* al fine di far visualizzare una url di invocazione nel dettaglio di una erogazione o fruizione che indirizzi il nodo corretto. La definizione corretta delle regole di proxy pass dipendono dall'architettura reale dei nodi in Load Balancing. Di seguito vengono forniti alcuni esempi al fine di esemplificare la funzionalità.

Ipotesi1: Nome del Canale corrisponde all'hostname di un nodo

In questo primo scenario ogni API sarà invocabile solamente su uno dei nodi che compongono il cluster (Fig. 8.20). Lo scenario prevede che gli identificativi dei canali corrispondo all'hostname di un nodo.

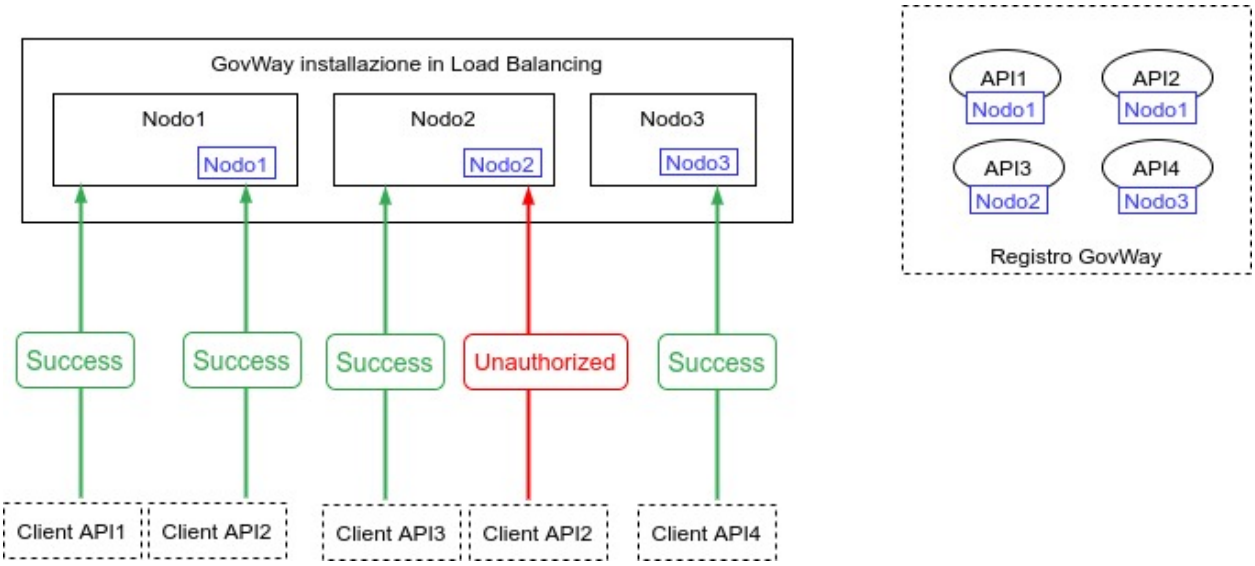


Fig. 8.20: Suddivisione delle API in Canali, scenario 1

Creando una regola di proxy pass (vedi *URL di Invocazione API*) con i seguenti criteri di Applicabilità:

- Espressione Regolare: true
- Regola: (.+)
- Profilo: API Gateway
- Soggetto: Qualsiasi
- Ruolo: Erogazione
- Tipo API: Qualsiasi

Nuova URL di Invocazione

- Base URL: http://${canale}/govway
- Contesto: ${0}

L'url di invocazione visualizzata per ogni erogazione indirizzerà il corretto host corrispondente al canale (Fig. 8.21):

- http://Nodo3/govway/ENTE/Api4/v1

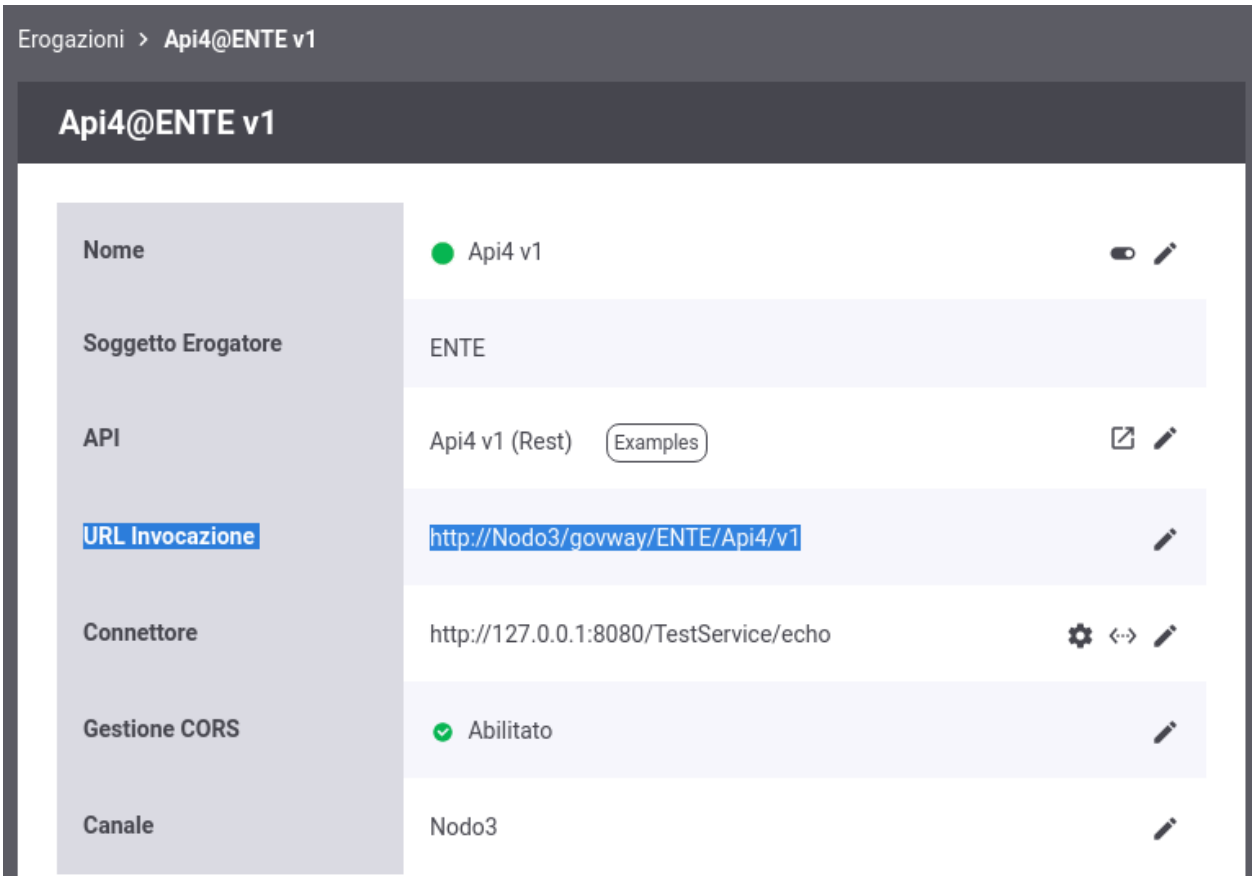


Fig. 8.21: Url di Invocazione, scenario 1

Ipotesi2: Nome del Canale corrisponde all'hostname di un load balancer

In questo scenario l'architettura è composta da 3 canali ognuno dei quali è gestito da due nodi GovWay bilanciati da un load balancer (Fig. 8.22). Lo scenario prevede che gli identificativi dei canali corrispondo all'hostname dei load balancer.

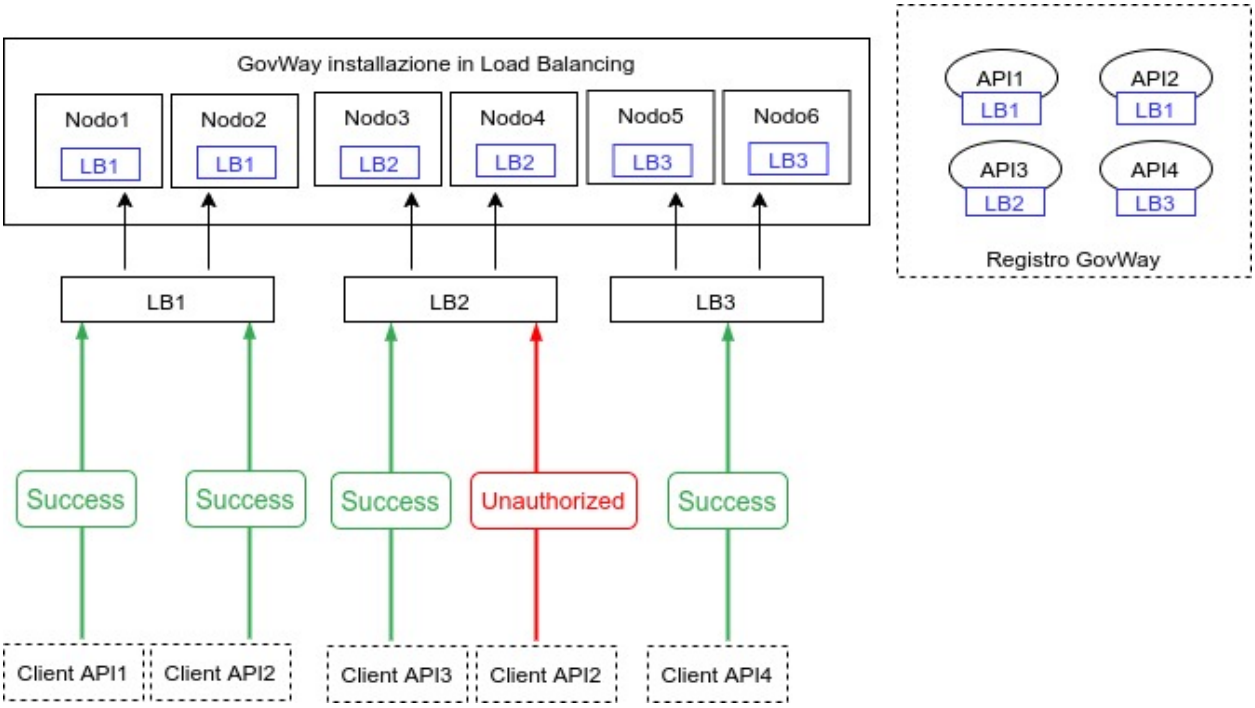


Fig. 8.22: Suddivisione delle API in Canali, scenario 2

La configurazione da utilizzare nelle regole di proxy pass è identica a quelle descritta nella Ipotesi 1

Ipotesi3: Nome del Canale corrisponde ad un contesto gestito da un FrontendWeb

In questo scenario ogni nodo in Load Balancing viene acceduto tramite un Frontend Web. Le richieste vengono redirette al corretto nodo in base ad una informazione di contesto presente nella url. Lo scenario prevede che gli identificativi dei canali corrispondo all'informazione di contesto utilizzato dal Frontend Web per inoltrare le richieste al corretto nodo.

Creando una regola di proxy pass (vedi *URL di Invocazione API*) con i seguenti criteri di Applicabilità:

- Espressione Regolare: true
- Regola: (.+)
- Profilo: API Gateway
- Soggetto: Qualsiasi
- Ruolo: Erogazione
- Tipo API: Qualsiasi

Nuova URL di Invocazione

- Base URL: http://frontend/govway
- Contesto: ${canale}${0}

L'url di invocazione visualizzata per ogni erogazione conterrà la corretta informazione di contesto che verrà utilizzata dal Frontend Web per smistare le richieste (Fig. 8.24):

- http://frontend/govway/C2/ENTE/Api3/v1

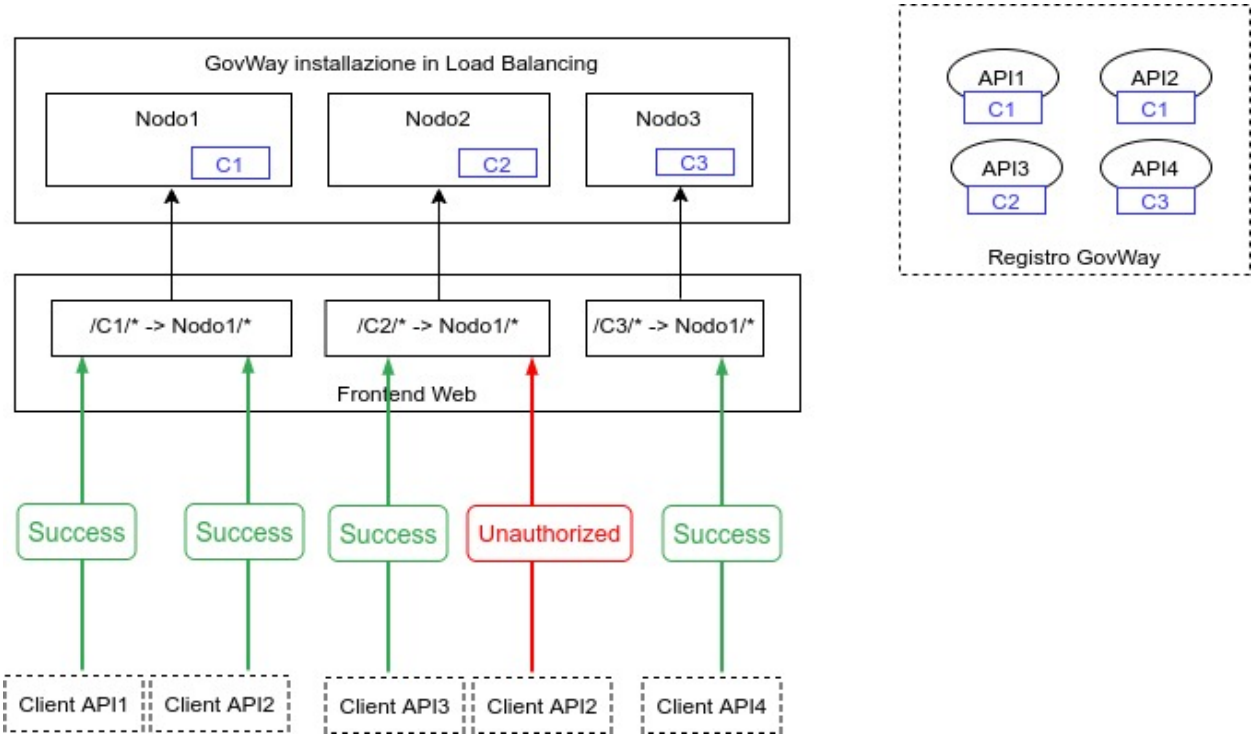


Fig. 8.23: Suddivisione delle API in Canali, scenario 3

8.1.7 Proprietà

La funzionalità consente di registrare una serie di proprietà che saranno aggiunte tra le proprietà java nel sistema tramite l'invocazione del metodo:

```
java.lang.System.setProperty(nome, valore);
```

La funzionalità è utilizzabile sia per impostare proprietà utilizzate direttamente da java, come ad es. le proprietà che riguardano il networking, sia per configurare altre funzionalità di GovWay, come ad es. *Gestione Proxy*.

8.2 Tracciamento

Accedendo la sezione *Configurazione* > *Tracciamento* si possono configurare i dettagli per la registrazione delle informazioni inerenti gli scambi sui servizi gestiti dal gateway. In particolare il gateway è in grado di memorizzare le seguenti tipologie di informazioni:

- *Transazioni*: tutte le proprietà inerenti il contesto di invocazione dei servizi (dati di indirizzamento, esito, tempi di elaborazione,...)
- *Messaggi Diagnostici*: tutte le informazioni necessarie per comprendere la fase di elaborazione delle richieste e indagare sulle anomalie occorse
- *Messaggi Applicativi*: salvataggio dei messaggi in transito sulle singole comunicazioni

In Fig. 8.26 è mostrata la pagina di configurazione del servizio di tracciamento.

Vediamo il significato delle sezioni di questa pagina:

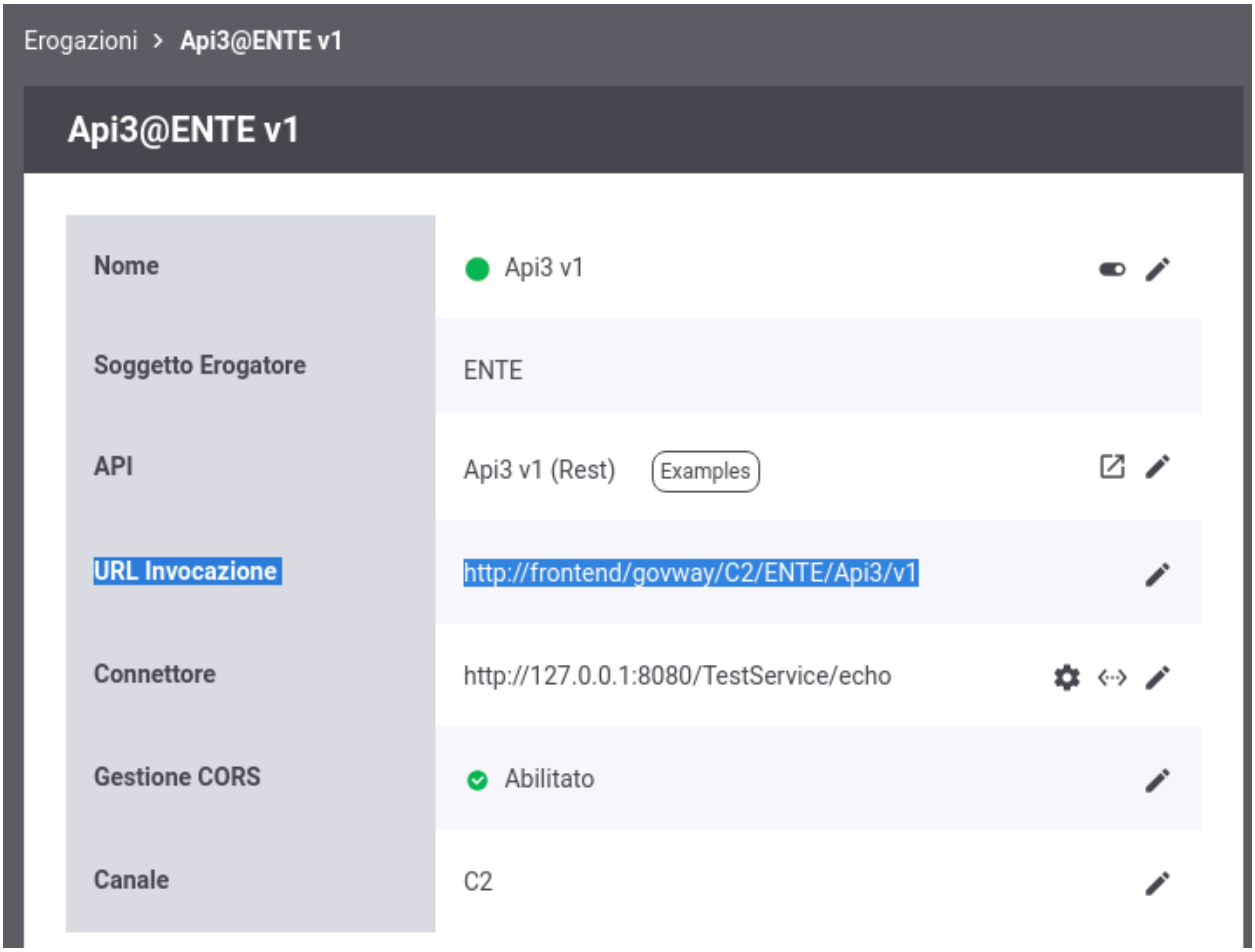


Fig. 8.24: Url di Invocazione, scenario 3

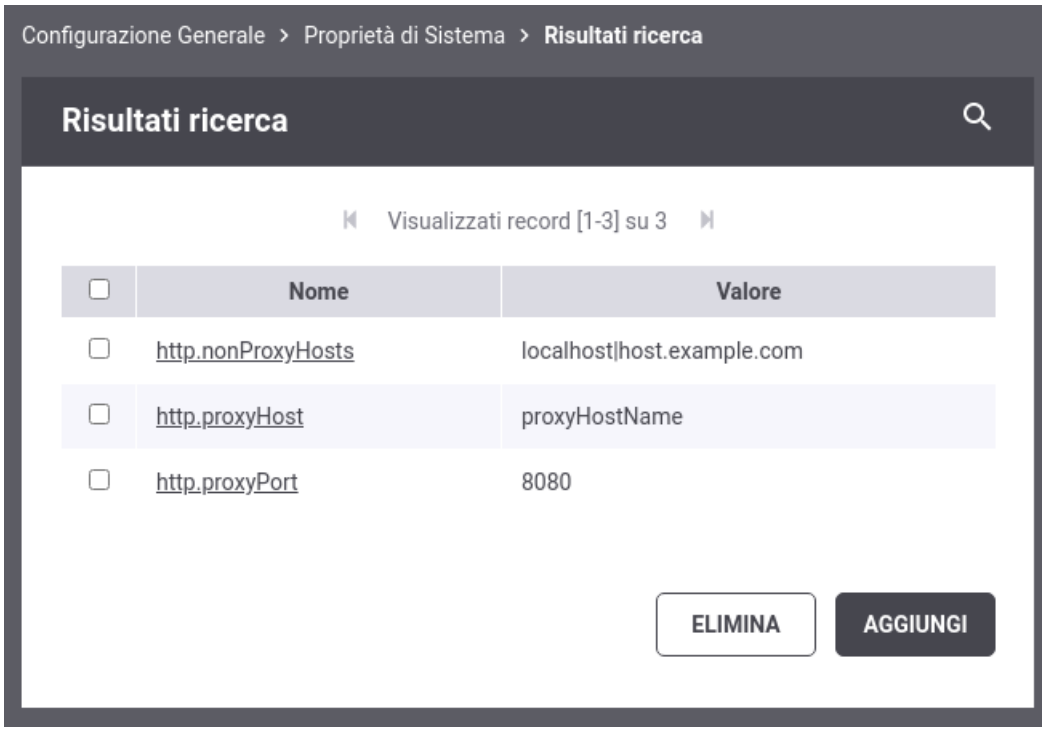


Fig. 8.25: Elenco di proprietà di una configurazione

- *Transazioni Registrate*: questa sezione consente di specificare quali transazioni memorizzare nell'archivio di monitoraggio in base all'esito rilevato in fase di elaborazione. Gli esiti sono suddivisi nei seguenti gruppi: Completate con successo, Fault applicativo, Fallite, Scartate, Violazione Policy Rate Limiting e Superamento Limite Richieste Complessive. Per ciascun esito è possibile abilitare o disabilitare la registrazione. È possibile inoltre, scegliendo l'opzione *Personalizzato* specificare puntualmente quali esiti di dettaglio includere.
- *Messaggi Diagnostici*: questa sezione consente di specificare il livello di verbosità dei messaggi diagnostici da generare. Si può distinguere il livello di verbosità per il salvataggio su *Database* e su *File*.
- *Registrazione Messaggi*: questa sezione consente di abilitare e configurare la registrazione dei messaggi in transito sul gateway durante l'elaborazione delle richieste e relative risposte. Una volta abilitata l'opzione si possono configurare i dettagli della funzionalità tramite il link *Configurazione*.

  Dalla sottosezione di configurazione si può distinguere il criterio di registrazione dei messaggi tra la Richiesta e la Risposta, abilitando/disabilitando solo la comunicazione desiderata. Sia per la Richiesta che per la Risposta, dopo aver optato per l'abilitazione della registrazione, si distingue tra:

  – *Ingresso*: il messaggio di richiesta o risposta nel momento in cui giunge sul gateway e quindi prima che venga sottoposto al processo di elaborazione previsto.

  – *Uscita*: il messaggio di richiesta o risposta nel momento in cui esce dal gateway, per raggiungere il nodo successivo del flusso, e quindi dopo che è stato sottoposto al processo di elaborazione previsto.

  Per ciascuno dei messaggi, su cui è stata abilitata la registrazione, è possibile scegliere quale elemento viene registrato:

  – *Headers*: vengono salvati gli header di trasporto (HTTP HEADERS) associati al messaggio.

  – *Body*: viene salvato il corpo del messaggio.

  – *Attachments*: vengono salvati gli eventuali attachments presenti nel messaggio.

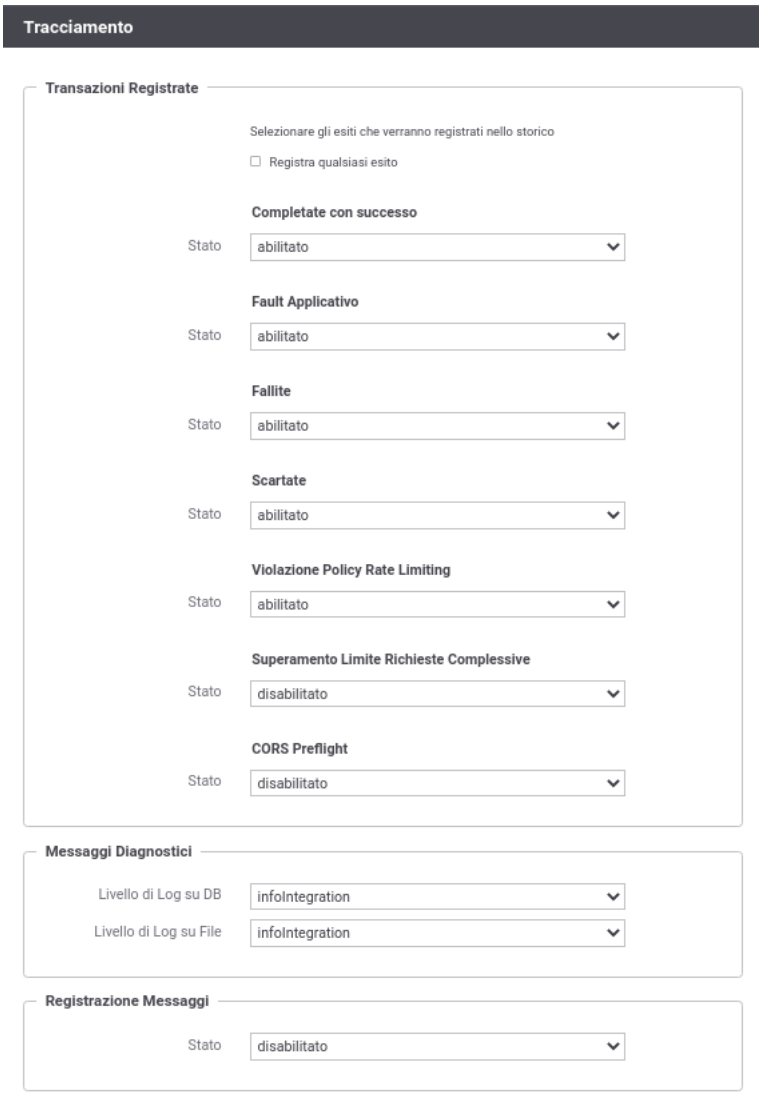


Fig. 8.26: Configurazione del servizio di tracciamento

Nota: Le configurazioni effettuate in questa sezione della console hanno valenza globale e quindi rappresentano il comportamento di default adottato dal gateway nella gestione dei diversi flussi di comunicazione. Tale comportamento può essere ridefinito puntualmente su ogni singola erogazione/fruizione agendo sulla voce di configurazione *Tracciamento* in quel contesto.

8.3 Controllo del Traffico

Accedendo la sezione *Configurazione* > *Controllo del Traffico* si possono impostare i parametri di configurazione relativamente alla fuzionalità che consente di stabilire le politiche di accesso alle risorse del gateway, nell'ottica di amministrare le risorse applicative a disposizione, ottimizzando le prestazioni e gestendo le situazioni di picco.

La configurazione della funzionalità di controllo del traffico (Fig. 8.27) si compone dei seguenti gruppi di configurazioni:

- *Limitazione Numero di Richieste Complessive*: consente di fissare un numero limite, riguardo le richieste gestibili simultaneamente dal gateway, bloccando le richieste in eccesso.
- *Controllo della Congestione*: consente di attivare il rilevamento dello stato di congestionamento del gateway, in seguito al superamento di una determinata soglia relativamente alle richieste simultanee.
- *Rate Limiting*: sezione per l'impostazione di policy al fine di attivare strategie di controllo del traffico con criteri di selezione specifici della singola richiesta.
- *Tempi Risposta*: sezione per l'impostazione dei valori limite relativi ai tempi di risposta dei servizi, sia nei casi di erogazione che di fruizione.

Le sezioni seguenti dettagliano questi elementi di configurazione.

8.3.1 Limitazione Numero di Richieste Complessive

Il primo livello di configurazione, presente nella pagina di accesso, consente di impostare i seguenti parametri:

- *Stato* (abilitato | disabilitato | warningOnly): Attiva il controllo sul numero di richieste simultanee in elaborazione. Selezionando l'opzione *abilitato* le richieste simultanee ricevute, che eccedono la soglia indicata (parametro *MaxRichiesteSimultanee*) verranno rifiutate restituendo al chiamante un errore. La tipologia di errore restituita è configurabile tramite l'elemento *Tipologia Errore* che appare solamente in caso di controllo abilitato.

  Il controllo sul numero di richieste simultanee in elaborazione può anche essere attivato in modalità *WarningOnly* dove, in caso il superamento della soglia, genera solamente un messaggio diagnostico di livello *error* e un evento che segnala l'accaduto.

- *Max Richieste Simultanee*: Corrisponde al numero massimo di richieste simultanee accettate. In genere è possibile fornire un valore accurato dopo aver valutato la portata massima del prodotto installato, in base alle risorse hardware disponibili e ai parametri di dimensionamento delle risorse applicative (ad esempio: numero connessioni al database, dimensioni della memoria java, ecc).

  Al superamento di tale valore non verranno accettate ulteriori richieste concorrenti, che verranno quindi rifiutate. Al verificarsi di questa situazione il gateway emette un evento specifico. Queste transazioni vengono marcate con esito *Superamento Limite Richieste* e saranno registrate solamente se previsto dalla configurazione (per default non vengono registrate). Per i dettagli sulla configurazione delle transazioni da registrare in base all'esito consultare la sezione *Tracciamento*.

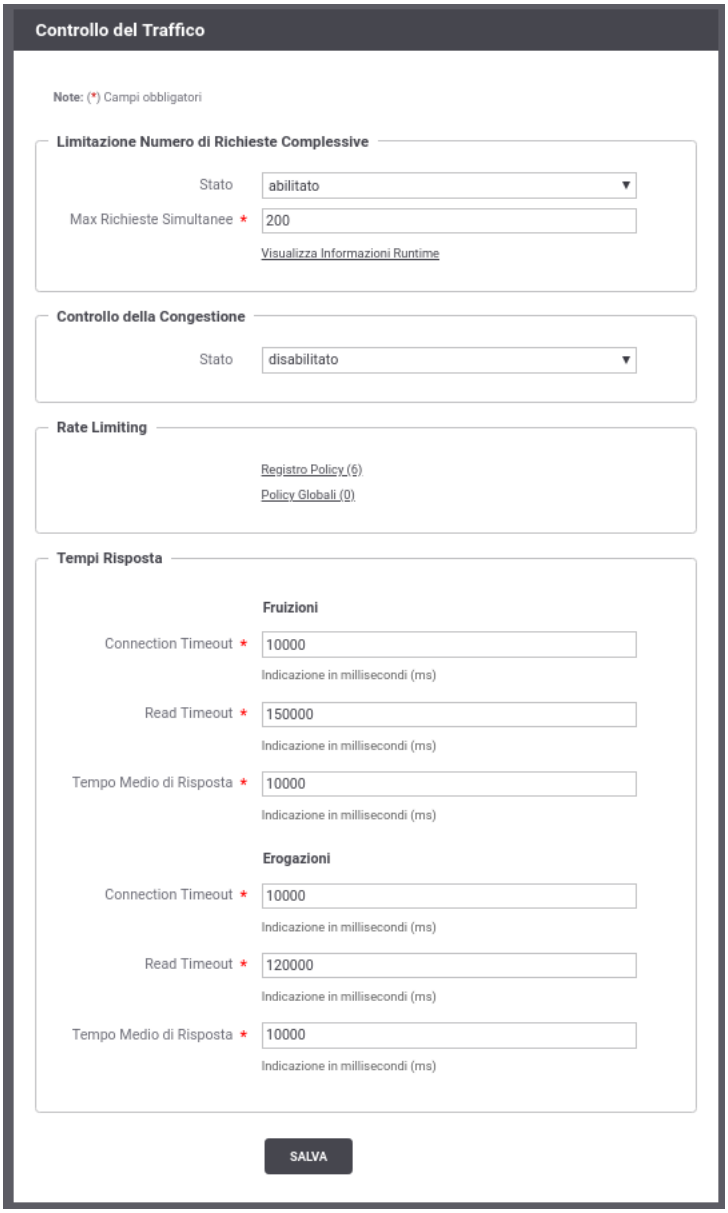


Fig. 8.27: Maschera per l'impostazione dei parametri di controllo del traffico

- *Tipo Errore per API SOAP* e *Includi Descrizione Errore* (Opzioni presenti solo con console in modalità avanzata): Imposta il tipo di errore restituito al chiamante nel caso di rifiuto dell'elaborazione per superamento della soglia del numero massimo di richieste simultanee. Le opzioni possibili sono le seguenti:
  - *Fault*: viene generato un messaggio di Fault contenente un codice ed una descrizione dell'errore rilevato nel caso l'elemento *Includi Descrizione Errore* sia abilitato, o un codice di errore generico altrimenti.
  - *Http 429 (Too Many Requests)*

    Http 503 (Service Unavailable)

    Http 500 (Internal Server Error)

    Viene generata una risposta HTTP con il codice selezionato, contenente una pagina html di errore, se l'elemento *Includi Descrizione Errore* è abilitato, o una risposta http vuota altrimenti.
- *Visualizza Informazioni Runtime*: Selezionando questo collegamento si apre una pagina (Fig. 8.28) che mostra in real-time le seguenti informazioni:
  - *Richieste Attive*: il numero di richieste simultanee attualmente in corso di elaborazione.
  - *Stato Gateway*: indica se il gateway ha raggiunto o meno lo stato di congestionamento, e quindi superata la soglia sul numero massimo di richieste simultanee.

Nota: L'indicatore è attivo solo nel caso in cui lo stato della successiva opzione *Controllo della Congestione* sia abilitato.

- *Refresh*: collegamento che consente di aggiornare le informazioni presentate nello schermo.

Fig. 8.28: Dati di congestionamento in tempo reale

8.3.2 Controllo della Congestione

Questa sezione consente di impostare i parametri relativi al controllo della congestione. Sono disponibili le seguenti opzioni:

- *Stato* (abilitato | disabilitato): Attiva il controllo sul numero di richieste simultanee al fine di individuare lo stato di congestionamento.
- *Soglia di Attivazione (%)*: Selezionando l'opzione *abilitato*, al passo precedente, questo elemento consente di indicare la soglia dello stato di congestionamento. La soglia da indicare è in percentuale rispetto al Numero Massimo Richieste Simultanee. Al superamento di tale soglia si entra nello stato di congestionamento conseguente emissione di un evento e un messaggio diagnostico al riguardo.

Nota:

Sulla base della percentuale indicata come soglia, una dicitura riporta nella pagina il valore di congestionamento calcolato in base al numero massimo di richieste simultanee.

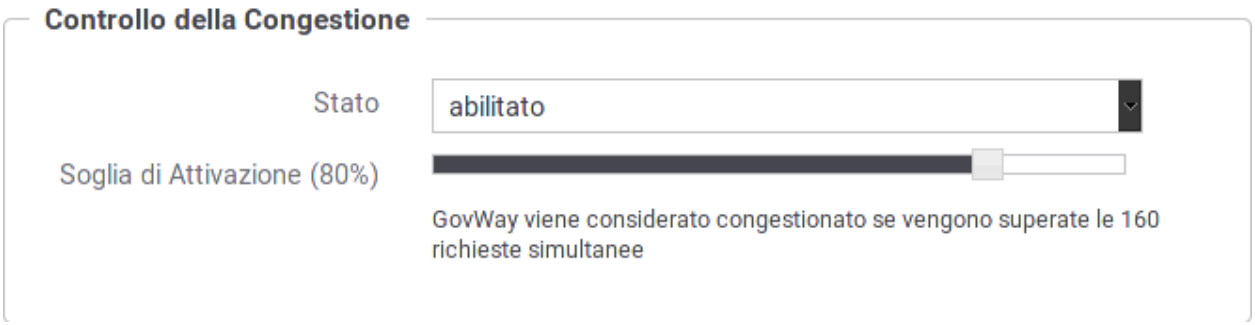


Fig. 8.29: Configurazione della soglia di congestionamento

8.3.3 Rate Limiting

Questa sezione consente di creare e attivare le policy di controllo del traffico. Gli elementi di configurazione presenti sono:

- *Tipo Errore per API SOAP* e *Includi Descrizione Errore* (Opzioni presenti solo con console in modalità avanzata): Imposta il tipo di errore restituito al chiamante nel caso venga rilevata una violazione delle policy configurate:
  - *Fault*: viene generato un messaggio di Fault contenente un codice ed una descrizione dell'errore rilevato nel caso l'elemento *Includi Descrizione Errore* sia abilitato, o un codice di errore generico altrimenti.
  - *Http 429 (Too Many Requests)*

    Http 503 (Service Unavailable)

    Http 500 (Internal Server Error)

    viene generata una risposta HTTP con il codice selezionato contenente una pagina html di errore se l'elemento *Includi Descrizione Errore* è abilitato, od una risposta http vuota altrimenti.
- *Registro Policy*: Consente di accedere al Registro delle Policy per visualizzare, modificare e creare le policy di controllo istanziabili per la configurazione del rate limiting. Tra parentesi viene visualizzato il numero di policy attualmente presenti nel registro. Questa funzionalità è descritta nella sezione *Registro Policy*.
- *Policy Globali*: Consente di accedere al Registro delle Policy Attivate in ambito globale, cioè operative sul traffico complessivo che transita sul gateway. A queste policy si aggiungono quelle eventualmente definite localmente nella configurazione specifica di ciascuna erogazione/fruizione. Tra parentesi viene visualizzato il numero di policy attualmente attivate. Questa funzionalità è descritta nella sezione *Policy Globali*.

8.3.4 Tempi Risposta

In questa sezione vengono indicati i valori limite di default riguardo i tempi di risposta dei servizi con cui il gateway interagisce durante l'elaborazione delle richieste. Nel caso delle erogazioni, si tratta dei tempi di risposta dei servizi interni, successivamente ad una richiesta di erogazione dall'esterno. Nel caso delle fruizioni, si tratta dei tempi di risposta dei servizi esterni, successivamente ad una richiesta di fruizione da parte di un client interno al dominio. I tempi configurabili sono:

- *Connection Timeout (ms)*: Intervallo di tempo atteso, sulle comunicazioni in uscita, prima di sollevare l'errore Connection Timeout (scadenza del tempo di attesa per stabilire una connessione).

- *Read Timeout (ms)*: Intervallo di tempo atteso, dopo aver stabilito una connessione in uscita, prima di sollevare l'errore di Read Timeout (scadenza del tempo di attesa per ricevere il payload dall'interlocutore).

Nota: È possibile impostare un timeout, in millisecondi, per la ricezione del payload della richiesta configurando la *Proprietà* "connettori.request.timeoutMs" sulla singola erogazione o fruizione.

- *Tempo Medio di Risposta (ms)*: Valore di soglia del tempo medio di risposta al fine di valutare la situazione di *Degrado Prestazionale*, condizione per l'applicabilità di eventuali politiche restrittive come documentate più avanti.

8.4 Rate Limiting

Questa sezione descrive come creare e attivare le policy di controllo del traffico:

- *Registro Policy*: Consente di accedere al Registro delle Policy per visualizzare, modificare e creare le policy di controllo istanziabili per la configurazione del rate limiting. Tra parentesi viene visualizzato il numero di policy attualmente presenti nel registro. Questa funzionalità è descritta nella sezione *Registro Policy*.
- *Policy Globali*: Consente di accedere al Registro delle Policy Attivate in ambito globale, cioè operative sul traffico complessivo che transita sul gateway. A queste policy si aggiungono quelle eventualmente definite localmente nella configurazione specifica di ciascuna erogazione/fruizione. Questa funzionalità è descritta nella sezione *Policy Globali*.

8.4.1 Registro Policy

Il Registro delle Policy è il repository dove si possono creare le policy di rate limiting che potranno essere successivamente istanziate. L'accesso alla sezione è possibile grazie all'omonimo collegamento presente nella sezione *Rate Limiting* della pagina principale del controllo del traffico.

La pagina indice del Registro delle Policy mostra l'elenco delle policy già presenti (Fig. 8.30).

Tramite il pulsante "Aggiungi" è possibile aprire la pagina di creazione di una policy di Rate Limiting (Fig. 8.31).

Descriviamo nel seguito i dati che è necessario inserire per la creazione di una policy. Si tenga presente che il sistema propone valori di default per alcuni campi; tali valori cambiano in base alle scelte operate sugli altri campi e possono essere considerati come "consigliati" in base alla combinazione di scelte attuate.

- *Policy*: In questa sezione sono presente i dati che identificano la policy.
  - *Nome*: Nome assegnato alla policy. Finché il campo non viene modificato dall'utente, viene proposto automaticamente un nome espressivo sulla base delle scelte operate sui rimanenti elementi del form.
  - *Descrizione*: Un testo di descrizione riferito alla policy. Finché il campo non viene modificato dall'utente, viene proposto un testo automatico di descrizione sulla base delle scelte operate sui rimanenti elementi del form.
  - *Metrica*: Si seleziona la metrica che la policy deve monitorare al fine di attuare le eventuali restrizioni. Sono disponibili le seguenti risorse:
    - *NumeroRichieste*: La policy effettua il controllo sul numero di richieste gestite. Selezionando questa risorsa si attiveranno i seguenti elementi per la configurazione dei valori di soglia:
      - *Modalità di Controllo*

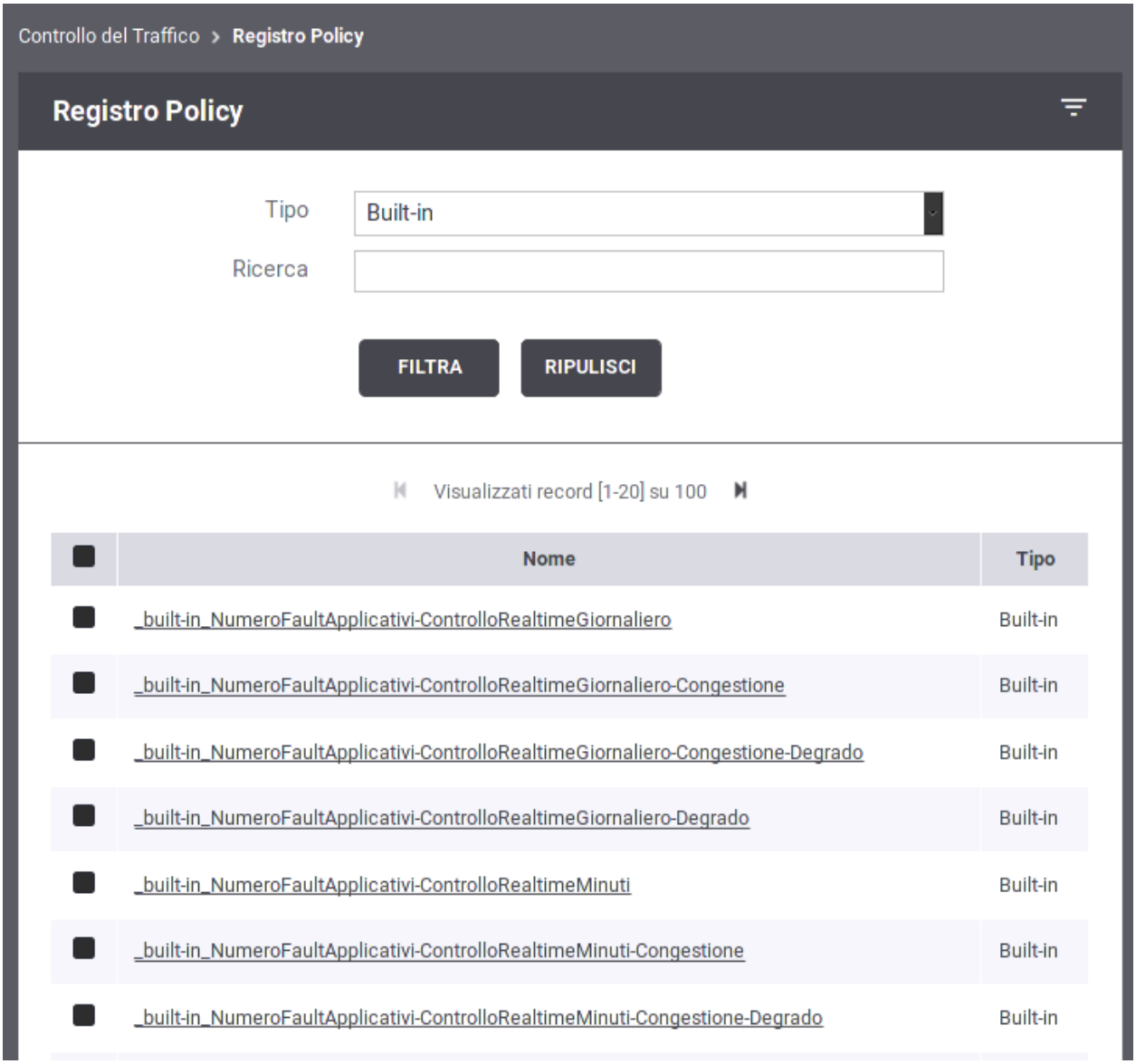


Fig. 8.30: Elenco delle Policy di Rate Limiting presenti nel registro

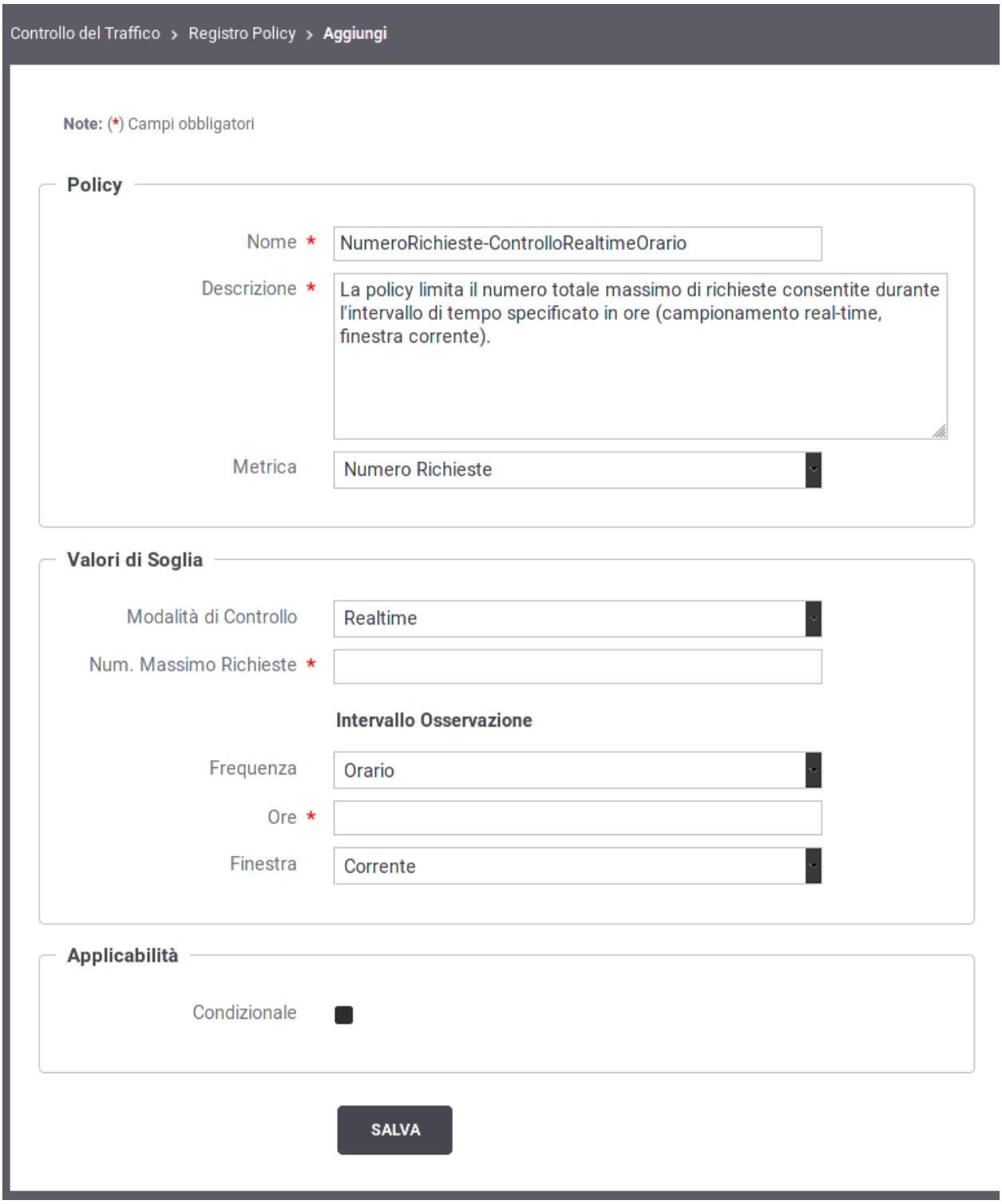


Fig. 8.31: Maschera per la creazione di una policy di Rate Limiting

- *Numero Massimo di Richieste*
- *Frequenza Intervallo Osservazione*
- *Intervallo Osservazione*
- *Finestra Osservazione*

* *NumeroRichiesteSimultanee*: La policy effettua il controllo sul numero di richieste simultanee gestite. Selezionando questa metrica si attiveranno i seguenti elementi per la configurazione dei valori di soglia:
  - *Numero Massimo di Richieste*
* *Dimensione Massima Messaggi*: La policy limita la dimensione massima, in KB, consentita per una richiesta e/o per una risposta. Selezionando questa metrica siattiveranno i seguenti elementi per la configurazione dei valori di soglia:
  - *Dimensione Richiesta*
  - *Dimensione Risposta*
* *OccupazioneBanda*: La policy effettua il controllo sulla banda occupata da e verso le comunicazioni con il gateway. Selezionando questa risorsa si attiveranno i seguenti elementi per la configurazione dei valori di soglia:
  - *Modalità di Controllo*
  - *Tipo Banda*
  - *Occupazione Massima di Banda (kb)*
  - *Frequenza Intervallo Osservazione*
  - *Intervallo Osservazione*
  - *Finestra Osservazione*
* *TempoComplessivioRisposta*: La policy controlla la quantità di tempo complessivamente impiegata dal gateway per la ricezione delle risposte dai servizi invocati. Selezionando questa metrica si attiveranno i seguenti elementi per la configurazione dei valori di soglia:
  - *Modalità di Controllo su Realtime (non modificabile)*
  - *Tipo Latenza*
  - *Occupazione Massima di Tempo (secondi)*
  - *Frequenza Intervallo Osservazione*
  - *Intervallo Osservazione*
  - *Finestra Osservazione*
* *TempoMedioRisposta*: La policy controlla il tempo medio impiegato dal gateway per la ricezione delle risposte dai servizi invocati. Selezionando questa metrica si attiveranno i seguenti elementi per la configurazione dei valori di soglia:
  - *Modalità di Controllo*
  - *Tipo Latenza*
  - *Tempo Medio Risposta (ms)*
  - *Frequenza Intervallo Osservazione*
  - *Intervallo Osservazione*

        - *Finestra Osservazione*

    * *NumeroRichiesteCompletateConSuccesso*

      NumeroRichiesteFallite

      NumeroFaultApplicativi

      La policy effettua il controllo sul numero di richieste gestite dal gateway e terminate con un esito che rientra nella casistica associata alla risorsa selezionata (completate con successo, fallite o fault applicativi). Selezionando questa metrica si attiveranno i seguenti elementi per la configurazione dei valori di soglia:

        - *Modalità di Controllo*
        - *Numero Massimo di Richieste*
        - *Frequenza Intervallo Osservazione*
        - *Intervallo Osservazione*
        - *Finestra Osservazione*

- *Valori di Soglia*: In questa sezione si specificano i valori di soglia (già anticipati al punto precedente), superati i quali, la policy risulta violata. Alcuni campi presenti in questa sezione cambiano in base alla risorsa monitorata.
  - *Simultanee*: Questa opzione è presente solo per la risorsa "NumeroRichieste". Attivandola si specifica che il criterio restrittivo entra in funzione al superamento di una soglia sul numero di richieste simultaneamente in gestione.
  - *Modalità di Controllo*: Rappresenta la modalità di raccolta dei dati di traffico che saranno usati per la valutazione della policy. Si può scegliere tra le seguenti opzioni:
    * *Realtime*: L'indicatore utilizzato per valutare la policy viene calcolato sulla base di dati raccolti in tempo reale durante l'elaborazione. Questa modalità assicura la massima accuratezza ma occorre tenere presenti le seguenti restrizioni nell'uso:
      1. I dati "realtime" vengono raccolti in maniera separata sui singoli nodi del cluster. Quindi il controllo effettuato dalla policy riguarderà il traffico sul singolo nodo.
      2. Si possono impostare criteri di controllo su grana temporale piccola: secondi, minuti, orario, giornaliero.
    * *Statistica*: L'indicatore utilizzato per valutare la policy viene calcolato sulla base delle informazioni statistiche presenti nel database di monitoraggio. L'accuratezza dei dati utilizzati per la valutazione è subordinata alla frequenza di aggiornamento dei dati statistici sul database. Inoltre tale modalità richiede il tracciamento delle transazioni sulle quali viene poi calcolata la statisticha (vedi sezione *Tracciamento*). In questa modalità:
      1. L'indicatore utilizzato per il confronto con la soglia della policy è sempre complessivo rispetto a tutti i nodi del cluster.
      2. Si possono impostare criteri di controllo con grana temporale ampia: orario, giornaliero, settimanale, mensile.
      3. Si può utilizzare la tipologia "finestra scorrevole" come valore per la "Finestra Osservazione", che descriveremo poco più avanti.
  - *Numero Massimo di Richieste*: Campo visibile solo per la metrica "NumeroRichieste". Consente di specificare la soglia per la policy. Quando il numero delle richieste, conteggiate secondo la logica specificata nella policy, supera questo valore, la policy risulta violata.

- *Tipo Banda*: Campo visibile solo per la metrica "OccupazioneBanda". Consente di specificare la modalità di calcolo della banda occupata per il confronto con la soglia impostata nella policy. Sono disponibili le seguenti opzioni:
  - *Banda Interna*: Ai fini del conteggio dell'occupazione di banda (in KB) verrà considerato il solo traffico relativo alle comunicazioni con gli applicativi interni al dominio.
  - *Banda Esterna*: Ai fini del conteggio dell'occupazione di banda (in KB) verrà considerato il solo traffico relativo alle comunicazioni con i servizi esterni al dominio.
  - *Banda Complessiva*: Ai fini del conteggio dell'occupazione di banda (in KB) verrà considerato tutto il traffico in entrata ed uscita sul gateway.
- *Occupazione Massima di Banda (kb)*: Campo visibile solo per la metrica "OccupazioneBanda". Consente di specificare la soglia per la policy. Quando la banda, calcolata secondo la logica specificata nella policy, supera questo valore, la policy risulta violata.
- *Tipo Latenza*: Campo visibile solo per le metriche "TempoComplessivoRisposta" e "TempoMedioRisposta". Consente di specificare la logica di calcolo del tempo di risposta sulla base delle due seguenti opzioni:
  - *Latenza Servizio*: Per il calcolo del tempo di risposta si considera unicamente il tempo di attesa del gateway dall'invio della richiesta alla ricezione della risposta.
  - *Latenza Totale*: Per il calcolo del tempo di risposta si considera, oltre alla latenza del servizio, anche il tempo di elaborazione del gateway dal momento dell'ingresso della richiesta fino all'uscita della risposta.
- *Occupazione Massima di Tempo (secondi)*: Campo visibile solo per la metrica "TempoComplessivoRisposta". Consente di specificare la soglia per la policy. Quando la latenza complessiva, calcolata secondo la logica specificata nella policy, supera questo valore, la policy risulta violata.
- *Tempo Medio Risposta (ms)*: Campo visibile solo per la metrica "TempoMedioRisposta". Consente di specificare la soglia per la policy. Quando la latenza media, calcolata secondo la logica specificata nella policy, supera questo valore, la policy risulta violata.
- *Frequenza Intervallo Osservazione*

  Intervallo Osservazione

  Finestra Osservazione

  La composizione di questi 3 campi specifica in quale intervallo temporale devono essere selezionati i dati da utilizzare per calcolare l'indicatore che deve essere confrontato con la soglia della policy.

  I valori di "Frequenza Intervallo Osservazione" e "Intervallo Osservazione" specificano la frequenza di campionamento dei dati utilizzati per la valutazione delle soglie. In particolare il valore da specificare come Intervallo Osservazione è sempre un numero intero (ad esempio inserendo 8 si campioneranno i dati su finestre di 8 secondi, 8 minuti, ecc, in base all'unità di misura indicata per la frequenza). Il valore selezionato come "Finestra» individua l'esatto intervallo utilizzato nella catena temporale ogni volta che si valuta la policy per una specifica richiesta di servizio.

  Per comprendere la logica con cui viene calcolata la finestra di osservazione è necessario introdurre il concetto di Data Attivazione Policy. Si tratta della data in cui la policy è stata applicata ad una richiesta in transito sul gateway. A partire da questa data vengono calcolate le finestre di osservazione in base alla frequenza di campionamento selezionata.

  In Fig. 8.32 è mostrato un confronto tra le diverse finestre di osservazione su un campionamento di 2 ore. La determinazione della finestra può essere analogamente trasposta su altre frequenze di campionamento.

  Riepilogando:

* *Corrente*: Indica che per il calcolo dell'indicatore saranno utilizzati i dati che rientrano nella finestra temporale in cui ricade la richiesta in esame.
* *Precedente*: Indica che per il calcolo dell'indicatore saranno utilizzati i dati che rientrano nella finestra temporale precedente a quella in cui ricade la richiesta in esame.
* *Scorrevole (disponibile solo nella Modalità Controllo "Statistica")*: Indica che per il calcolo dell'indicatore saranno utilizzati i dati che rientrano in una finestra dinamica che ha come estremo superiore l'ora piena subito precedente all'istante della richiesta in fase di valutazione.

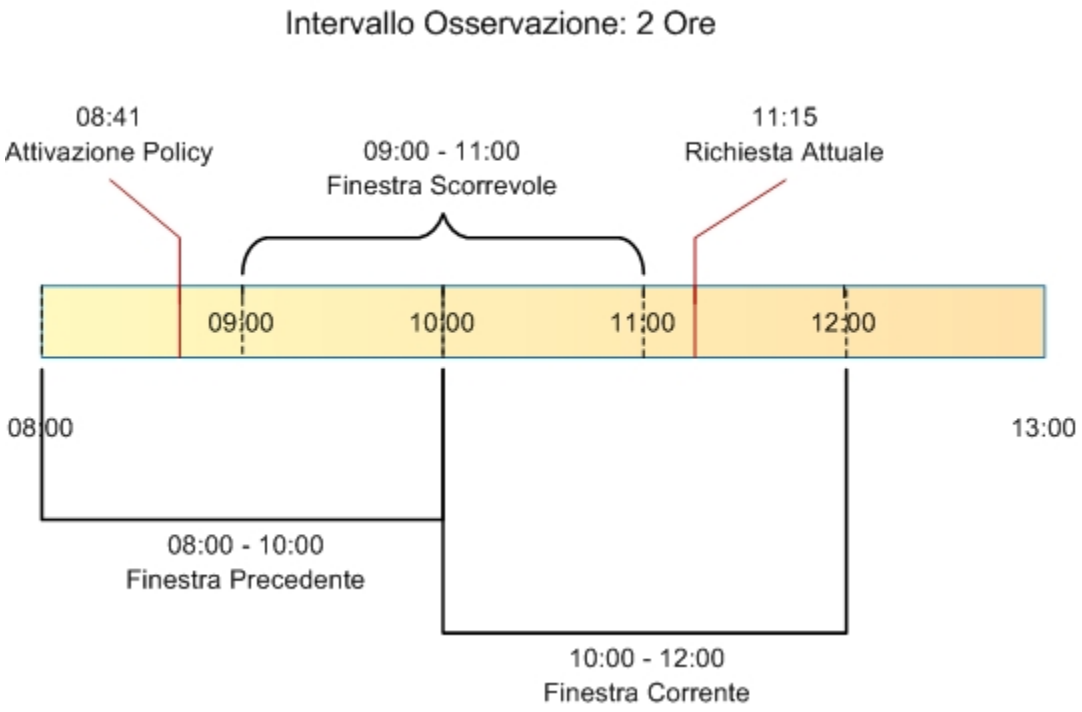


Fig. 8.32: Finestre di osservazione su un campionamento di 2 ore

- *Applicabilità*: Questa sezione della policy consente di restringere l'applicabilità della policy sulla base di alcuni criteri (Fig. 8.33). Sono presenti i seguenti campi:
  - *Condizionale*: Se questa opzione non è attiva, la policy si applica in maniera incondizionata. Attivando l'opzione, la policy risulterà applicabile sulla base dei criteri specificati nei campi successivi.
  - *In presenza di Congestione del Traffico*: Attivando questa opzione la policy risulta applicabile solo quando sussiste lo stato di congestionamento. Affinché questo evento venga rilevato è necessario che sia abilitato il "Controllo della Congestione", descritto in precedenza, e che risulti superata la soglia impostata sul numero di richieste simultanee.
  - *In presenza di Degrado Prestazionale*: Attivando questa opzione, la policy risulta applicabile solo in caso si rilevi un degrado prestazionale sullo specifico servizio corrispondente alla richiesta in gestione sul gateway. Per la rilevazione del degrado prestazionale si utilizzano le soglie "Tempo Medio di Risposta" impostate sia per le fruizioni che per le erogazioni. Come descritto in precedenza, tali soglie vengono definite per default nella sezione "Configurazione > Controllo del Traffico", ma possono essere ridefinite al livello del singolo connettore. Per il calcolo del tempo medio di risposta del servizio, da confrontare con la soglia impostata, si utilizza il criterio definito con i campi seguenti:
    * *Modalità di Controllo*
    * *Tempo Medio Risposta*

- *Frequenza Intervallo Osservazione*
- *Intervallo Osservazione*
- *Finestra Osservazione*

Per tutti questi campi valgono le medesime descrizioni già riportate nella sezione precedente "Valori di Soglia".

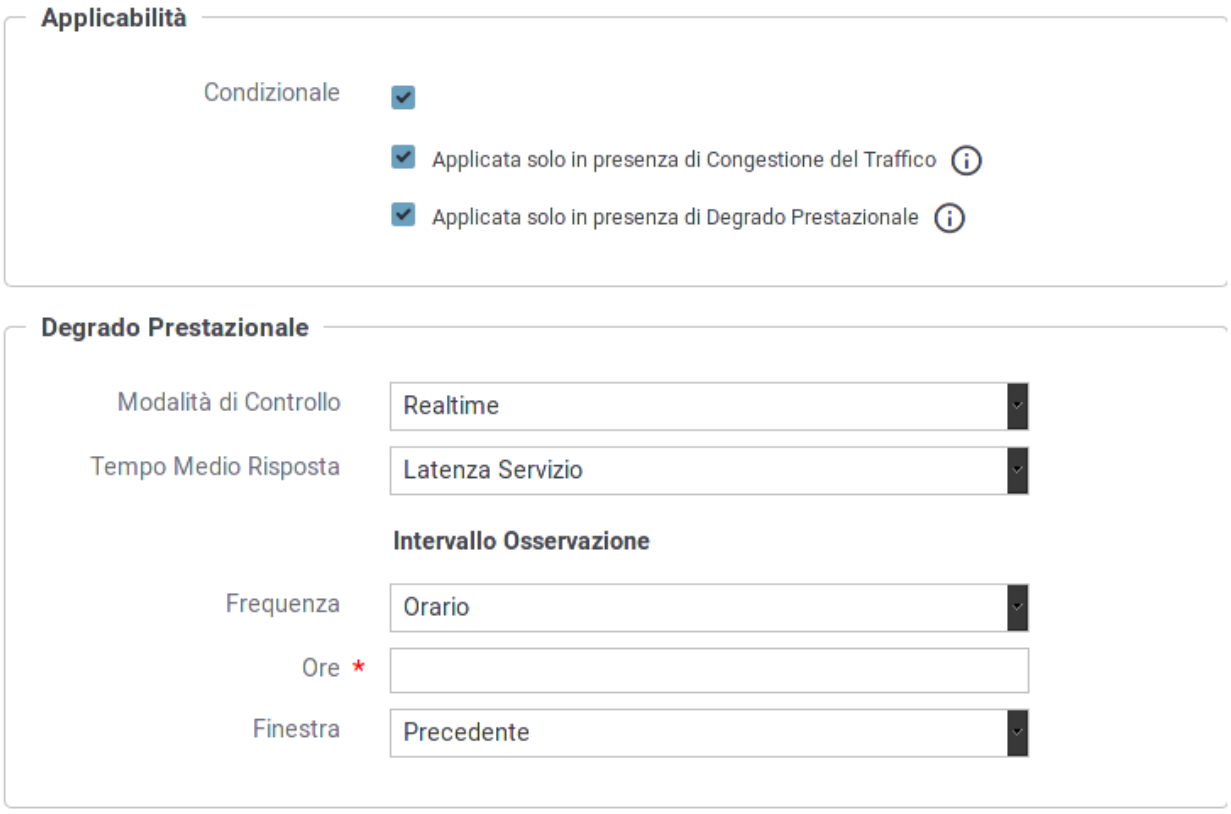


Fig. 8.33: Opzioni per l'applicabilità di una policy di rate limiting

Nota: Se si selezionano più opzioni di applicabilità queste si considerano connesse secondo l'operatore logico AND.

8.4.2 Policy Globali

Questa sezione consente di definire le policy di rate limiting che hanno un raggio d'azione che supera la singola erogazione/fruizione ed effettua quindi valutazioni su un campo più ampio.

L'attivazione di una Policy Globale segue in prevalenza il medesimo criterio già descritto nella sezione *Registrazione di una policy* riguardo il caso della configurazione di una singola erogazione/fruizione. Vi sono però alcune differenze che riguadano i criteri di raggruppamento, per il calcolo dei valori di soglia, e i criteri di filtro per l'applicabilità della policy.

Raggruppamento

Come descritto nella sezione *Registrazione di una policy* è possibile definire dei criteri di raggruppamento che consentono di verificare i valori di soglia. La logica è del tutto analoga a quella dell'operatore GROUP BY del linguaggio SQL.

I criteri di raggruppamento, per una policy a livello globale, sono maggiori rispetto a quelli descritti in precedenza (Fig. 8.34):

- *Fruizione/Erogazione*
- *Soggetto Erogatore*
- *API*
- *Azione/Risorsa*
- *Soggetto Fruitore*
- *Applicativo Fruitore*
- *Token*
- *Raggruppamento per Chiave*

Fig. 8.34: Definizione criteri di raggruppamento per la policy di rate limiting

Filtro

Abilitando questa sezione è possibile indicare i criteri affinché la policy sia applicabile in base alle caratteristiche di ciascuna richiesta in ingresso. In assenza di filtro, la policy sarà valutata su tutte le richieste in ingresso che riguardano l'erogazione/fruizione che si sta configurando. I criteri di filtro, per una policy a livello globale, sono maggiori rispetto a quelli descritti in precedenza nella sezione *Registrazione di una policy* (Fig. 8.35):

- *Stato*: Opzione per abilitare/disabilitare il filtro.
- *Ruolo Gateway*: Opzione per filtrare le richieste di servizio in base al ruolo ricoperto dal gateway nella specifica richiesta: Fruitore o Erogatore.
- *Profilo*: Opzione per filtrare le richieste di servizio in base al profilo di utilizzo del Gateway. Nel caso si sia selezionata un singolo profilo (o se il gateway ne supporta uno solo) viene visualizzato il valore attuale in modo non modificabile.
- *Ruolo Erogatore*: Opzione per filtrare le richieste di servizio in base al ruolo posseduto dal soggetto erogatore. Tramite la lista è possibile selezionare uno tra i ruoli censiti nel registro. La selezione di un ruolo esclude la possibilità di selezionare un soggetto erogatore.
- *Soggetto Erogatore*: Opzione per filtrare le richieste di servizio in base al soggetto erogatore. Tramite la lista è possibile selezionare uno tra i soggetti censiti nel registro. La selezione di un soggetto esclude la possibilità di selezionare un ruolo erogatore.
- *API*: Opzione per filtrare le richieste in base alla API invocata. Tramite la lista è possibile selezionare una tra le API censite nel registro. Se è stato selezionato un soggetto erogatore, saranno elencati solo le API da esso erogate. Analogamente, se è stato selezionato un profilo, saranno elencate solo API relative a tale profilo.
- *Azione/Risorsa*: Opzione per filtrare le richieste di servizio in base all'azione/risorsa invocata. Tramite la lista è possibile selezionare una tra le azioni/risorse censite nel registro. Se è stato selezionato una API, saranno elencati solo le azioni ad essa appartenenti.
- *Ruolo Fruitore*: Opzione per filtrare le richieste di servizio in base al ruolo posseduto dal soggetto fruitore. Tramite la lista è possibile selezionare uno tra i ruoli censiti nel registro. La selezione di un ruolo esclude la possibilità di selezionare un soggetto fruitore.
- *Soggetto Fruitore*: Opzione per filtrare le richieste di servizio in base al soggetto fruitore. Tramite la lista è possibile selezionare uno tra i soggetti censiti nel registro. Se è stato selezionato un servizio, saranno elencati solo i soggetti fruitori del medesimo. La selezione di un soggetto esclude la possibilità di selezionare un ruolo fruitore.
- *Applicativo Fruitore*: Opzione per filtrare le richieste di servizio in base all'applicativo fruitore (opzione non disponibile nel caso di una erogazione). Tramite la lista è possibile selezionare uno tra i servizi applicativi censiti nel registro. Se sono stati selezionati servizi e/o soggetti, la lista presentata sarà filtrata di conseguenza.
- *Filtro per Chiave*: Si tratta di un'opzione avanzata che consente di filtrare le richieste in ingresso sul gateway in base ad una chiave che può essere specificata in maniera personalizzata. Questa parte è già stata descritta in maniera approfondita nella sezione *Registrazione di una policy*.

Nota: È possibile specificare più di un criterio di filtro; la logica applicata sarà quella dell'operatore AND.

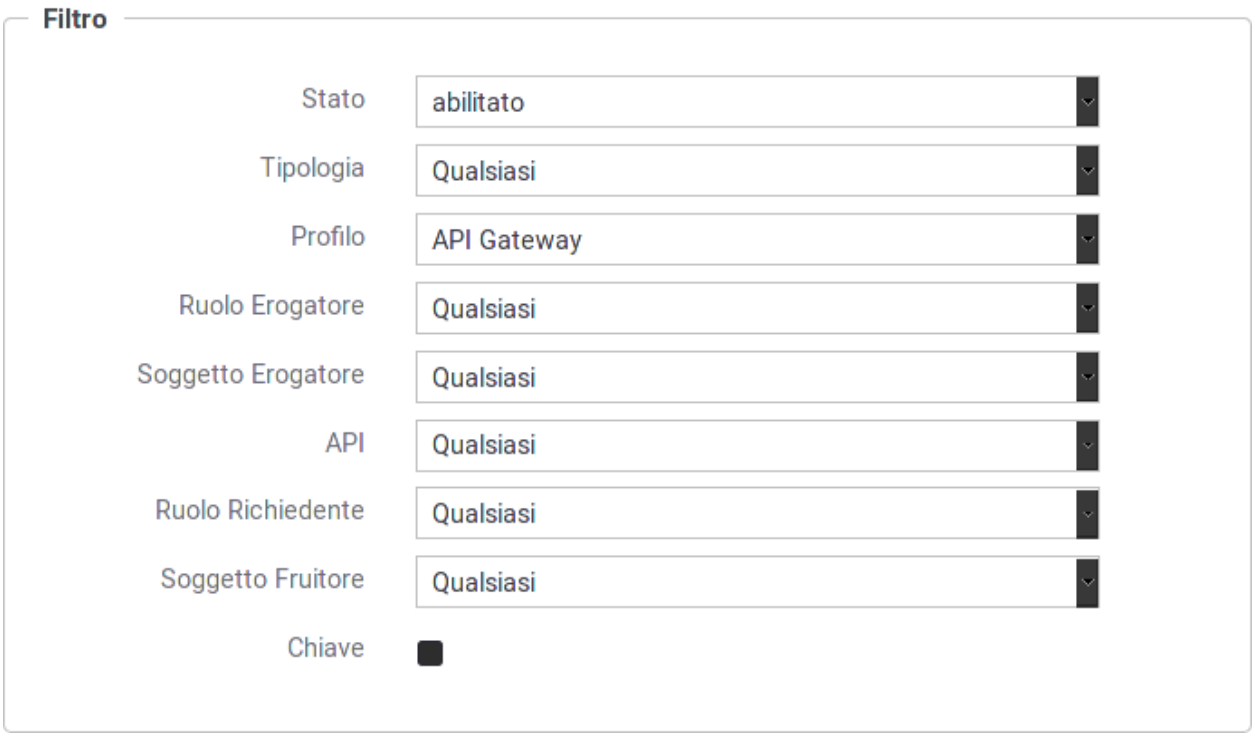


Fig. 8.35: Definizione del filtro per l'istanza della policy di rate limiting

8.4.3 Visualizzazione Statistiche Policy

Quando una policy è attivata si ha la possibilità di accedere ad una finestra che fornisce una sintesi dei dati statistici legati all'applicazione della policy in fase di controllo del traffico.

Per visualizzare questa finestra è sufficiente accedere all'elenco delle policy attivate ed utilizzare il collegamento "Visualizza" nella colonna "Runtime" (Fig. 8.36).

Si noti che saranno visualizzati dei dati solo dopo la data di attivazione della policy e cioè dopo che è transitata la prima richiesta cui viene applicata la policy.

I dati statistici riportati sono i seguenti:

- *Criterio di Collezionamento dei dati*: I criteri di raggruppamento utilizzati dalla policy.
- *Dati Generali*:
  - *Il numero istantaneo delle richieste attive*
  - *la data di attivazione della policy (che corrisponde alla data di primo utilizzo della medesima)*
- *Dati collezionati per la risorsa <nomeRisorsa>*: dati di sintesi sulle transazioni cui è stata applicata la policy.

Sono inoltre disponibili i seguenti collegamenti:

- *Refresh*: per aggiornare i dati visualizzati.
- *Reset Contatori*: per azzerare i valori visualizzati (solo nella modalità di controllo realtime).

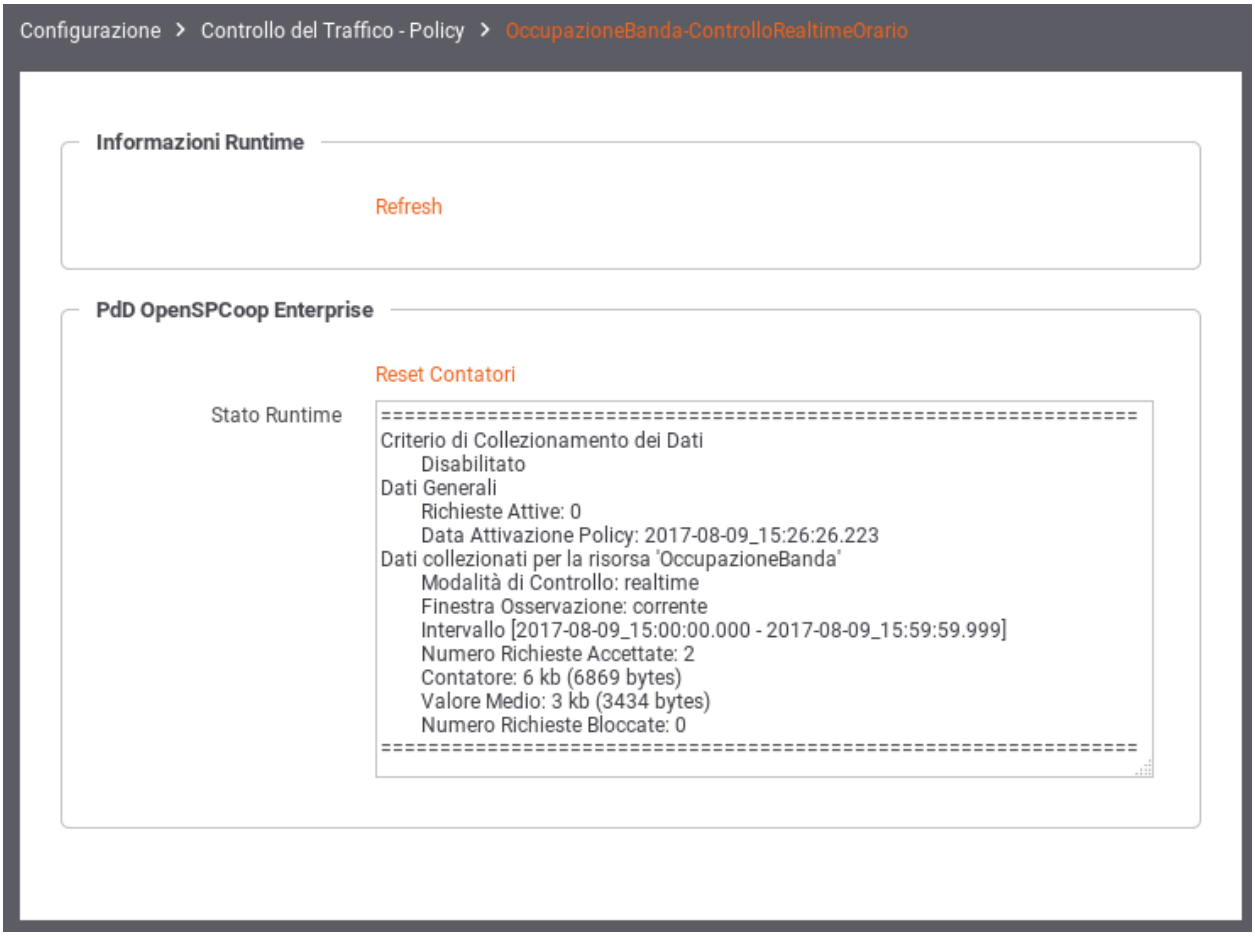


Fig. 8.36: Dati statistici relativi ad una policy di rate limiting

8.4.4 Filtro o Raggruppamento Personalizzato

Nella sezione *Registrazione di una policy* è possibile utilizzare dei criteri di raggruppamento per il valore di soglia o un filtro di applicabilità personalizzato in modo da definire un comportamento specifico per le proprie esigenze di servizio. Una configurazione personalizzata richiede la realizzazione di un plugin che contiene la logica di filtro e/o il raggruppamento personalizzato; il plugin consiste nell'implementazione di una classe java che implementa l'interfaccia:

```
package org.openspcoop2.pdd.core.controllo_traffico.plugins;
public interface IRateLimiting {
    public String estraiValoreFiltro(Logger log,Dati datiRichiesta) throws␣
↪PluginsException;
    public String estraiValoreCollezionamentoDati(Logger log,Dati␣
↪datiRichiesta) throws PluginsException;
}
```

La classe realizzata deve essere successivamente registrata tramite una entry da aggiungere all'interno del file (da creare se non esiste) */etc/govway/govway_local.classRegistry.properties* di GovWay:

```
org.openspcoop2.pdd.controlloTraffico.rateLimiting.<tipo>=<fully qualified␣
↪class name>
```

La stringa <tipo> diventa utilizzabile come "Tipo Personalizzato" da indicare in fase di configurazione per un criterio di filtro personalizzato (Fig. 8.37) e/o per un criterio di raggruppamento personalizzato (Fig. 8.38).

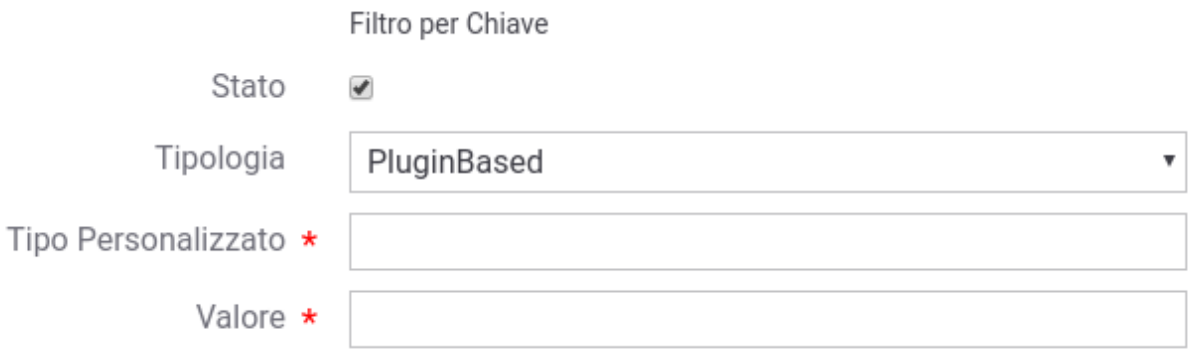


Fig. 8.37: Filtro Personalizzato

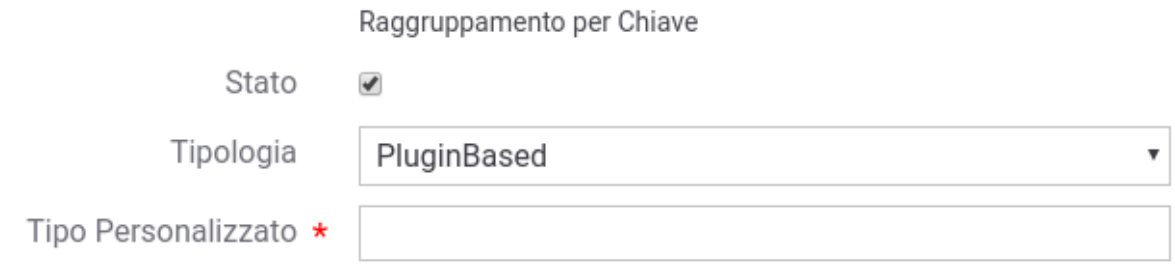


Fig. 8.38: Raggruppamento Personalizzato

8.5 Token Policy

Per poter definire politiche di controllo degli accessi basate sui Bearer Token o per poterne spedire uno verso l'endpoint associato ad un connettore è necessario creare delle Token Policy da riferire nelle configurazioni degli specifici servizi. La gestione delle Token Policy si effettua andando alla sezione *Configurazione* > *Token Policy* della govwayConsole. Per creare una nuova policy si utilizza il pulsante *Aggiungi*. Il form di creazione appare inizialmente come quello illustrato in Fig. 8.43.

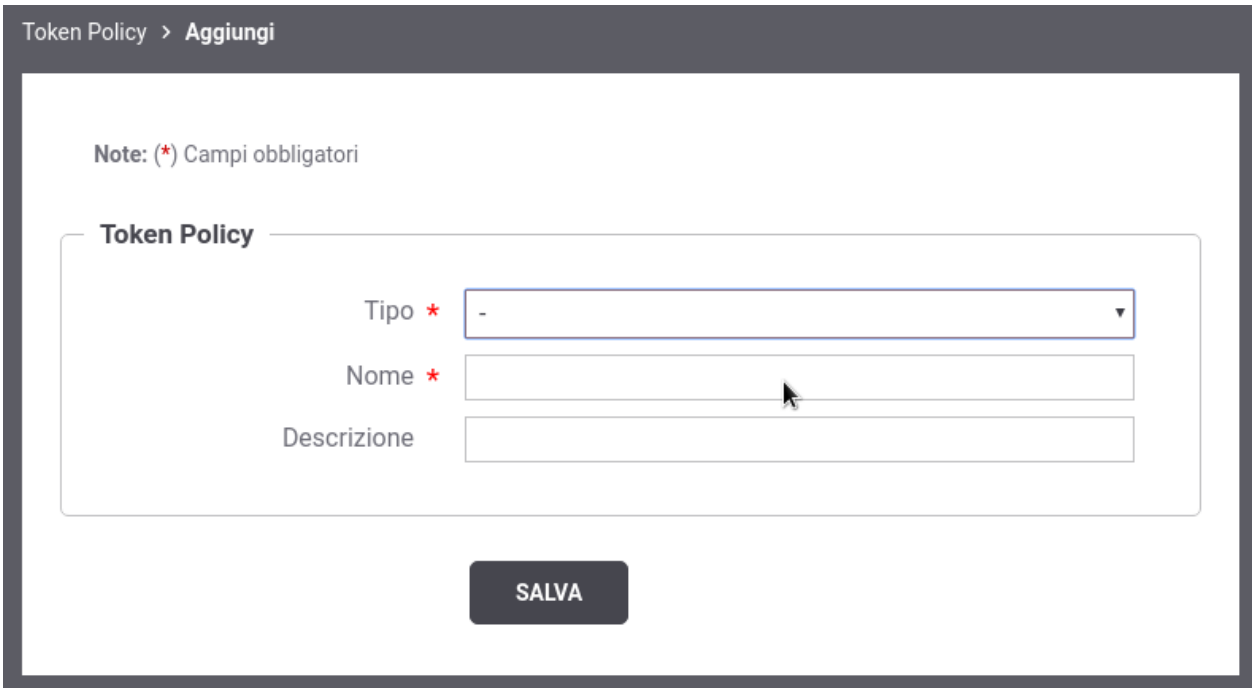


Fig. 8.39: Creazione di una Token Policy

Inizialmente si inseriscono i dati identificativi:

- *Nome*: nome univoco da assegnare alla policy
- *Tipo*: determina il tipo di policy:
  - *Validazione*: definisce una policy utilizzabile per validare Bearer Token nel Controllo degli Accessi (*Autenticazione Token*)
  - *Negoziazione*: definisce i criteri per la negoziazione di un Bearer Token poi utilizzato sui connettori nei quali sarà associata la policy (*Autenticazione Token*)
- *Descrizione*: testo di descrizione generale della policy

I parametri richiesti differiscono a seconda del tipo selezionato. Le sezioni successive dettagliano i due tipi supportati.

8.5.1 Token Policy Negoziazione

Per poter definire politiche che consentono di spedire un Bearer Token verso l'endpoint associato ad un connettore è necessario creare delle Token Policy da riferire nelle configurazioni degli specifici servizi. La gestione delle Token Policy si effettua andando alla sezione *Configurazione* > *Token Policy* della govwayConsole. Per creare una nuova policy si utilizza il pulsante *Aggiungi*. Il form di creazione appare inizialmente come quello illustrato in Fig. 8.43.

Inizialmente si inseriscono i dati identificativi:

- *Nome*: nome univoco da assegnare alla policy
- *Tipo*: deve essere selezionato il tipo *Negoziazione*
- *Descrizione*: testo di descrizione generale della policy

Al passo successivo si inseriscono le Informazioni Generali. Nella sezione *Token Endpoint* si specifica il tipo di negoziazione e i vari parametri necessari:

- *Tipo*: indica la modalità di negoziazione del token. I valori possibili sono:
  - *Client Credentials*: modalità di negoziazione "Client Credentials Grant" descritta nel RFC 6749 (https://tools.ietf.org/html/rfc6749#page-40);
  - *Resource Owner Password Credentials*: modalità di negoziazione "Resource Owner Password Credentials Grant" descritta nel RFC 6749 (https://tools.ietf.org/html/rfc6749#page-37);
  - *Signed JWT*: modalità di negoziazione "Client Credentials Grant" descritta nella sezione 2.2 del RFC 7523 (https://datatracker.ietf.org/doc/html/rfc7523#section-2.2) che prevede lo scambio di un'asserzione JWT firmata tramite certificato x.509 con l'authorization server;
  - *Signed JWT with Client Secret*: modalità di negoziazione identica alla precedente dove però l'asserzione JWT viene firmata tramite una chiave simmetrica.
- *URL*: endpoint del servizio di negoziazione token.
- *Connection Timeout*: Tempo massimo in millisecondi di attesa per stabilire una connessione con il server di negoziazione token.
- *Read Timeout*: Tempo massimo in millisecondi di attesa per la ricezione di una risposta dal server di negoziazione token.
- *Https*: Parametri di configurazione nel caso in cui il server di negoziazione token richieda un accesso Https.
- *Proxy*: Parametri di configurazione nel caso in cui il server di negoziazione token richieda l'uso di un proxy per l'accesso.

Nel caso sia attivato il flag relativo ad un Proxy o una configurazione Https saranno presentate delle sezioni omonime dove poter inserire i dati di configurazione richiesti.

I parametri di configurazioni relativi al tipo di negoziazione del token configurato vengono descritti nelle sezioni "*Client Credentials / Resource Owner Password Credentials*" e "*Signed JWT*".

Nella sezione "Dati Richiesta" potranno invece essere definiti ulteriori criteri che riguardano la richiesta di un token:

- *Scope*: elenco di scope utente richiesti;
- *Audience*: audience per il quale si vorrebbe ottenere il token;
- *Parametri*: consentene di indicare per riga ulteriori parametri (nome=valore) da inserire nella richiesta.

Infine nella sezione "Token Forward" si può configurare la modalità di inoltro del token verso l'endpoint del connettore a cui verrà associata la policy che stiamo definendo:

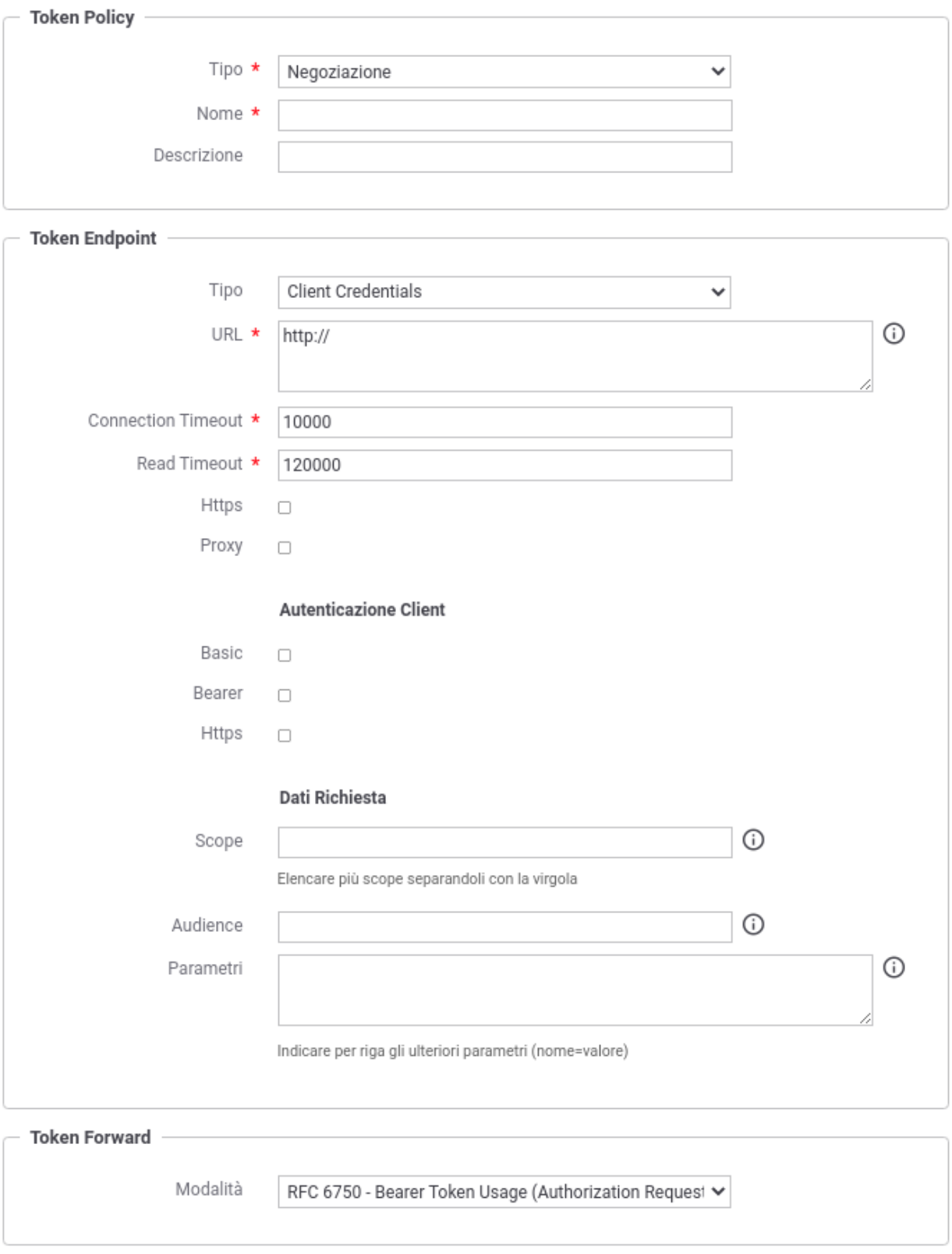


Fig. 8.40: Informazioni generali di una Token Policy

- *RFC 6750 - Bearer Token Usage (Authorization Request Header Field)*: Il token viene inoltrato al destinatario utilizzando l'header Authorization presente nella richiesta HTTP.
- *RFC 6750 - Bearer Token Usage (URI Query Parameter)*: Il token viene inoltrato al destinatario tramite parametro access_token della Query String.
- *Header HTTP*: Il token viene inoltrato al destinatario utilizzando un header HTTP il cui nome deve essere specificato nel campo seguente.
- *Parametro URL*: Il token viene inoltrato al destinatario utilizzando un parametro della Query String il cui nome deve essere specificato nel campo seguente.

Nelle sezioni successive vengono forniti i dettagli relativi alle modalità di negoziazione di un token nel caso sia previsto un jwt firmato o meno.

Client Credentials / Resource Owner Password Credentials

In entrambe le modalità è necessario definire i parametri di configurazione richiesti dall'authorization server per autenticare GovWay come client autorizzato a negoziare il token. Le modalità supportate sono le seguenti:

- *Autenticazione Http Basic*: flag da attivare nel caso in cui il servizio di negoziazione richieda autenticazione di tipo HTTP-BASIC. In questo caso dovranno essere forniti Client-ID e Client-Secret nei campi successivi. È inoltre possibile indicare se la coppia di credenziali deve essere codificata nella richiesta "x-www-form-urlencoded" oppure deve essere inserita in un header HTTP "Authorization".
- *Autenticazione Bearer*: flag da attivare nel caso in cui il servizio di negoziazione richieda autenticazione tramite un bearer token. Il token dovrà essere indicato nel campo successivo fornito.
- *Autenticazione Https*: flag da attivare nel caso in cui il servizio di negoziazione richieda autenticazione di tipo Https. In questo caso dovranno essere forniti tutti i dati di configurazione nei campi presenti nella sezione "https".

Se il tipo di negoziazione selezionato è "Resource Owner Password Credentials", si dovra inoltre fornire i dati di configurazione specifici dell'autenticazione utente:

- *Username* e *Password*: Dovranno essere forniti Username e Password dell'utente per cui verrà effettuata la negoziazione del token.

Signed JWT

Nel caso di modalità di negoziazione basata su uno scambio di un JWT firmato con l'authorization server si dovranno fornire tutti i parametri che andranno a definire il JWT firmato.

Innanzitutto se la modalità prevede una firma tramite chiave asimmetrica devono essere forniti i parametri di accesso ad un keystore contenente la chiave privata da utilizzare per la firma:

- *Tipo*: tipo del keystore selezionabile tra JKS, PKCS12, JWK o uno dei tipi di keystore PKCS11 registrati ("*Device PKCS11*");
- *File*: path assoluto al keystore;
- *Password*: password del keystre;
- *Alias Chiave Privata* e *Password Chiave Privata*: alias con cui è stata registrata la chiave nel keystore e eventuale password.

Nella sezione "JWT Signature" si deve indicare l'algoritmo di firma e l'eventuale chiave segreta nel caso in cui sia prevista una firma tramite chiave simmetrica.

All'interno della sezione "JWT Header" si possono definire quali parametri dovranno essere presenti nella parte non firmata dell'asserzione JWT:

- *Key Id (kid)*: indicazione della chiave utilizzata per attuare la firma dell'asserzione JWT, in una delle seguenti modalità:
  - Alias Chiave Privata (solamente in caso di firma con chiave asimmetrica): nel claim "kid" viene impostato l'alias della chiave privata indicato nella precedente sezione di configurazione;
  - Client ID: viene utilizzato il medesimo valore associato al claim "client_id" inserito nel payload firmato del JWT;
  - Personalizzato: permette di indicare un valore qualsiasi anche formato da parti dinamiche risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*).
- *X.509 Certificate* (solamente in caso di firma con chiave asimmetrica):
  - "x5c": viene inserito il certificato utilizzato per firmare l'asserzione JWT;
  - "x5u": viene indicata una url dove reperire il certificato di firma.
- *Digest X.509 Certificate* (solamente in caso di firma con chiave asimmetrica): consente di indicare il digest del certificato di firma nella modalità SHA1 (x5t) o SHA256 (x5t#S256);
- *Type*: valore inserito nel claim "typ";
- *Content Type*: se abilitato, il claim "cty" verrà valorizzato con il content-type associato alla richiesta effettuata all'authorization server.

Nella sezione "JWT Payload" si devono definire i parametri inseriti nella parte firmata dell'asserzione JWT:

- *Client ID*: identificativo del client censito sull'AuthorizationServer che verrà indicato nel claim "client_id" dell'asserzione JWT;
- *Audience*: identifica l'authorization server come destinario dell'asserzione JWT (claim "aud");
- *Issuer*: dominio del soggetto firmatario dell'asserzione; se non viene fornito un valore il claim "iss" verrà valorizzato con il nome del soggetto associato al dominio di gestione della richiesta;
- *Subject*: il claim "sub" verrà valorizzato con la medesima informazione inserita nel claim "client_id" se nel campo Subject non viene fornito alcun valore;
- *Time to Live*: indica la validità temporale, in secondi, a partire dalla data di creazione dell'asserzione;
- *Claims*: consente di inserire ulteriori claims nel payload JWT firmato, indicandoli per riga nel formato "nome=valore". Fornendo un valore che inizia e termina con le parentesi graffe si definisce un oggetto json, come ad esempio:
  - claimTest={«prova»:»valoreProva», «prova2»:»${header:X-Example}»}

  Se il valore inizia e termina con le parentesi quadre si definisce invece un array json, come ad esempio:
  - claimTest=[«valoreProva», «valoreProva2», «${header:X-Example}»]

Tutti i valori definiti nella sezione "JWT Payload" possono contenere parti dinamiche che verranno risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*). Inoltre se non si generare un determinato claim è possibile utilizzare la keyword "${undefined}" come valore del campo.

Signed JWT (PDND)

La modalità di negoziazione basata su uno scambio di un JWT firmato con l'authorization server, descritta nella sezione "*Signed JWT*", è stata selezionata come modalità di negoziazione dei token sulla Piattaforma Digitale Nazionale Dati (PDND).

Il protocollo di negoziazione, oltre ai parametri standard previsti dal rfc 7523 (https://datatracker.ietf.org/doc/html/rfc7523#section-2.2), prevede l'inserimento di alcuni parametri aggiuntivi all'interno del payload del JWT firmato:

- *purposeId*: rappresente l'identificativo della finalità dell'accordo di adesione recuperabile dalla piattaforma PDND;
- *sessionInfo*: informazioni di sessione che non vengono gestite sulla piattaforma PDND ma consentono di essere inviate al momento della negoziazione per poi essere riportate all'interno dell'access token generato dalla PDND.

Entrambi i parametri sono configurabili attivando la modalità PDND successivamente alla selezione del tipo "Signed JWT" come modalità di negoziazione (Fig. 8.41) e (Fig. 8.42).

Fig. 8.41: Modalità di negoziazione "Signed JWT" via PDND

Fig. 8.42: Modalità di negoziazione "Signed JWT" via PDND: purposeId e sessionInfo

Infine anche nella sezione “Dati Richiesta”, nel caso venga attivata la modalità *PDND*, viene aggiunto un ulteriore parametro richiesto dal protocollo PDND: “client_id”. Se non viene definito alcun valore per il parametro verrà utilizzato il medesimo valore associato al Client ID definito nel payload del JWT.

8.5.2 Token Policy Validazione

Per poter definire politiche di controllo degli accessi basate sui Bearer Token è necessario creare delle Token Policy da riferire nelle configurazioni degli specifici servizi. La gestione delle Token Policy si effettua andando alla sezione *Configurazione* > *Token Policy* della govwayConsole. Per creare una nuova policy si utilizza il pulsante *Aggiungi*. Il form di creazione appare inizialmente come quello illustrato in Fig. 8.43.

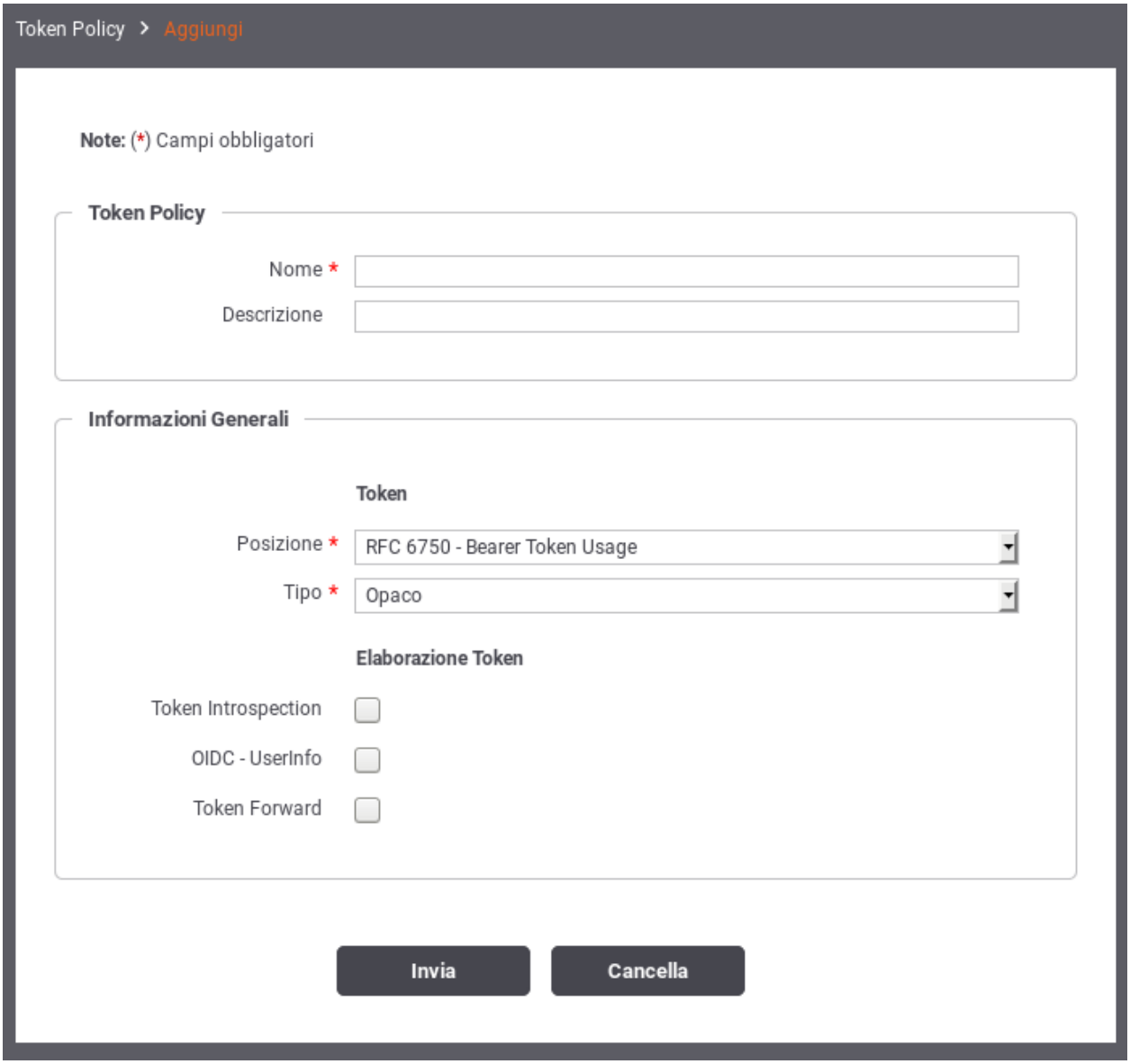


Fig. 8.43: Informazioni generali di una Token Policy

Inizialmente si inseriscono i dati identificativi:

- *Nome*: nome univoco da assegnare alla policy
- *Tipo*: deve essere selezionato il tipo *Validazione*
- *Descrizione*: testo di descrizione generale della policy

Al passo successivo si inseriscono le Informazioni Generali. Nella sezione *Token* si specifica il tipo di token accettato e il metodo di passaggio:

- *Posizione*: indica la modalità di passaggio del token da parte dell'applicativo richiedente. I valori possibili sono:
  - *RFC 6750 - Bearer Token Usage*: la modalità di passaggio del token è una qualsiasi delle tre previste dallo standard RFC 6750 (le tre opzioni successive a questa).
  - *RFC 6750 - Bearer Token Usage (Authorization Request Header Field)*: la modalità di passaggio del token è quella che prevede l'inserimento nell'header «Authorization» del messaggio di richiesta. Ad esempio:

```
GET /resource HTTP/1.1
Host: server.example.com
Authorization: Bearer mF_9.B5f-4.1JqM
```

  - *RFC 6750 - Bearer Token Usage (Form-Encoded Body Parameter)*: la modalità di passaggio del token è quella di inserirlo nel body della richiesta, eseguita con una POST, utilizzando il parametro *access_token*, come ad esempio:

```
POST /resource HTTP/1.1
Host: server.example.com
Content-Type: application/x-www-form-urlencoded

access_token=mF_9.B5f-4.1JqM
```

  - *RFC 6750 - Bearer Token Usage (URI Query Parameter)*: la modalità di passaggio del token è quella di utilizzare il parametro *access_token* della Query String, come ad esempio:

```
GET /resource?access_token=mF_9.B5f-4.1JqM HTTP/1.1
Host: server.example.com
```

  - *Header HTTP*: la modalità di passaggio del token è quella di inserirlo in un header http custom, il cui nome deve essere fornito nel campo *Nome Header Http*, che appare di seguito.
  - *Parametro URL*: la modalità di passaggio del token è quella di inserirlo in un parametro custom della query string. Il nome del parametro deve essere fornito nel campo *Nome Parametro URL*, che appare di seguito.
- *Tipo*: specifica il tipo di token che il gateway attende di ricevere. I valori possibili sono:
  - *JWS*: un JSON Web Token di tipo «Signed».
  - *JWE*: un JSON Web Token di tipo «Encrypt».
  - *Opaco*: un generico token di tipo non specificato.

Nella sezione *Elaborazione Token* si specificano le azioni che si possono compiere durante la fase di elaborazione del token ricevuto. Le opzioni disponibili sono:

- Validazione JWT
- Token Introspection
- OIDC - UserInfo
- Token Forward

Le sezioni successive dettagliano questi elementi.

Validazione JWT

Nel caso in cui il token sia di tipo JWT (quindi JWE o JWS), questa opzione attiva la validazione basata su tale standard (Fig. 8.44).

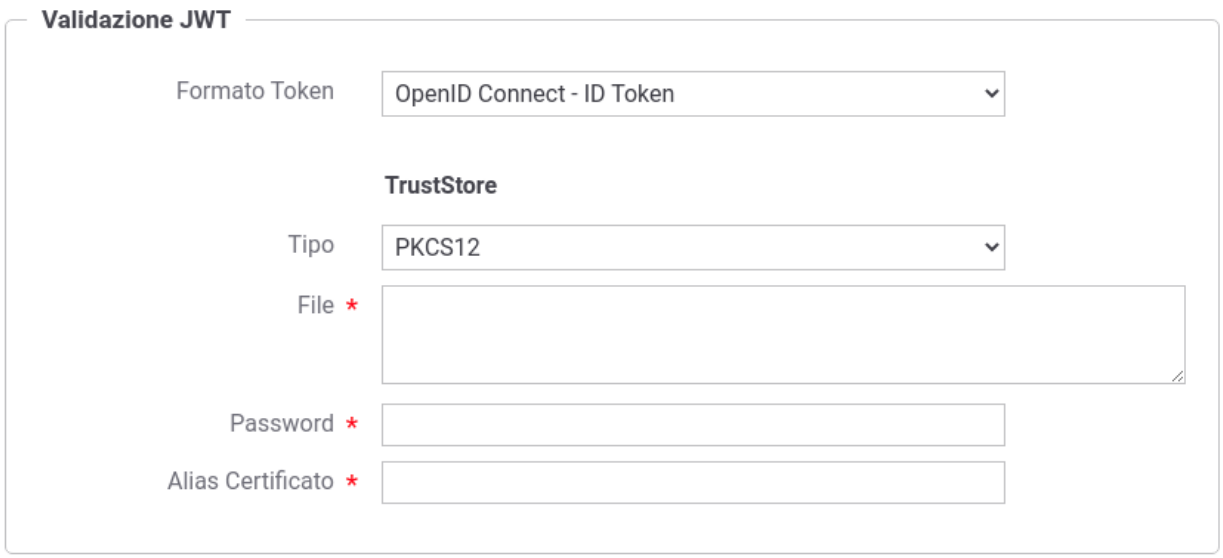


Fig. 8.44: Dati di configurazione della validazione JWT

I dati da inserire sono:

- *Formato Token*: indica il formato atteso del payload contenuto nel token JWT. Maggiori dettagli sul mapping vengono forniti in "*Formati dei token*". I valori possibili sono:
  - *RFC 7519 - JSON Web Token*: claims attesi definiti nel RFC "https://datatracker.ietf.org/doc/html/rfc7519#section-4";
  - *OpenID Connect - ID Token*: definiti nel RFC "https://openid.net/specs/openid-connect-core-1_0.html#IDToken";
  - *Google - ID Token*: claims definiti in "https://developers.google.com/identity/protocols/oauth2/openid-connect#obtainuserinfo";
  - *Personalizzato*: consente di definire un mapping puntuale tra il nome di un claim e l'informazione che GovWay cerca di estrarre dal token (Fig. 8.48);
  - *Plugin*: consente di indicare il nome di una classe che implementa una logica di parsing personalizzata (deve implementare l'interfaccia "org.openspcoop2.pdd.core.token.parser.ITokenParser").
- *KeyStore*: I parametri di configurazione del keystore da utilizzare per il servizio di validazione.

Token Introspection

Questa sezione consente di attivare la validazione del token ricevuto attraverso un servizio di Token Introspection i cui dati di accesso devono essere forniti in questo contesto (Fig. 8.45).

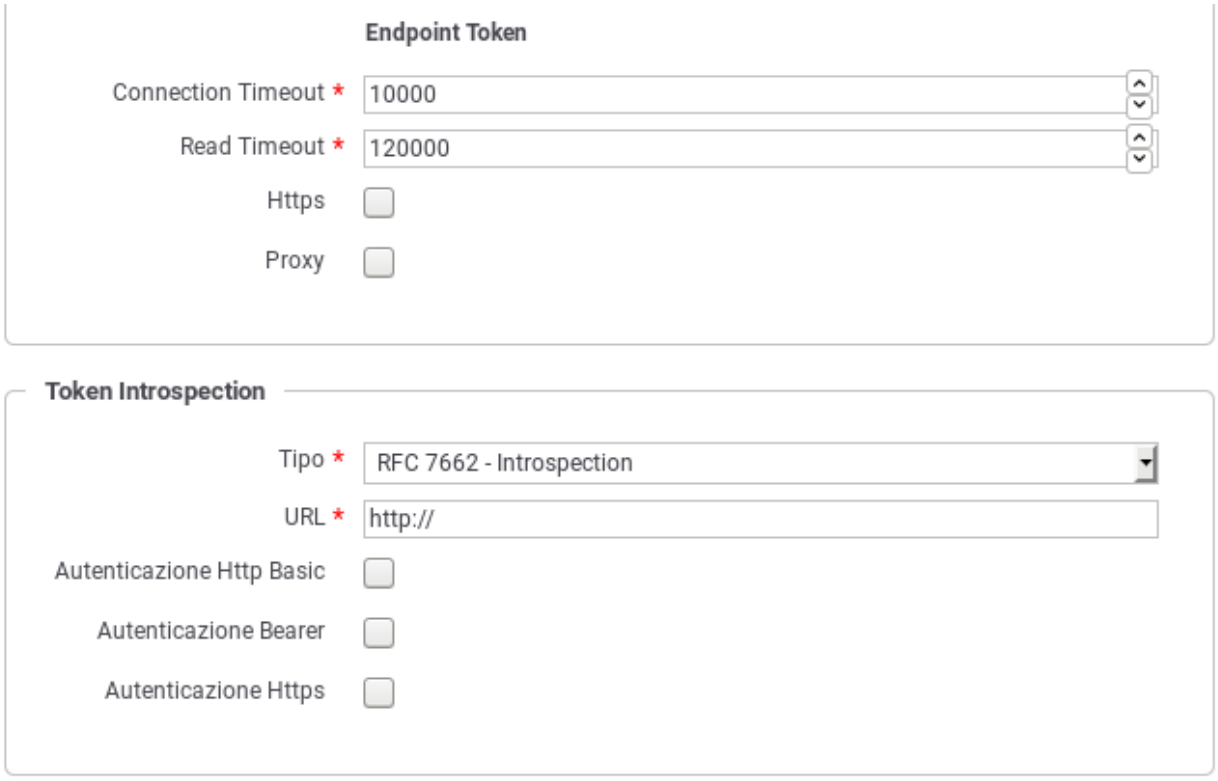


Fig. 8.45: Dati di puntamento al servizio di Token Instrospection

Per il corretto puntamento al servizio di Token Introspection devono essere forniti in prima istanza i parametri generali legati all'endpoint riferito:

- *Connection Timeout*: Tempo massimo in millisecondi di attesa per stabilire una connessione con il server di validazione token.
- *Read Timeout*: Tempo massimo in millisecondi di attesa per la ricezione di una risposta dal server di validazione token.
- *Https*: Parametri di configurazione nel caso in cui il server di validazione token richieda un accesso Https.
- *Proxy*: Parametri di configurazione nel caso in cui il server di validazione token richieda l'uso di un proxy per l'accesso.

Successivamente devono essere forniti i dati di configurazione specifici del servizio di Token Introspection:

- *Tipo*: tipologia del servizio. A scelta tra i seguenti valori:
  - *RFC 7662 - Introspection*: Servizio di introspection conforme allo standard RFC 7662 "https://datatracker.ietf.org/doc/html/rfc7662". Richiede che vengano forniti i seguenti dati:
    - *URL*: endpoint del servizio di introspection.

    - *Autenticazione Http Basic*: flag da attivare nel caso in cui il servizio di introspection richieda autenticazione di tipo HTTP-BASIC. In questo caso dovranno essere forniti Username e Password nei campi successivi.
    - *Autenticazione Bearer*: flag da attivare nel caso in cui il servizio di introspection richieda autenticazione tramite un token. Il token dovrà essere indicato nel campo successivo fornito.
    - *Autenticazione Https*: flag da attivare nel caso in cui il servizio di introspection richieda autenticazione di tipo Https. In questo caso dovranno essere forniti tutti i dati di configurazione nei campi presenti nella sezione "https".
- *Google - TokenInfo*: Riferimento al servizio di token introspection di Google. L'unico campo da fornire in questo caso è la URL del servizio. Il sistema precompila questo campo con il valore di default *https://www.googleapis.com/oauth2/v3/tokeninfo*.
- *Personalizzato*: Questa opzione consente di configurare un servizio di Token Introspection personalizzato (Fig. 8.46) attraverso i seguenti dati:
    - *URL*: la URL del servizio di introspection;
    - *Autenticazione*: consente di configurare, selezionando il flag opportuno, il tipo di autenticazione richiesta dal servizio di introspection personalizzato;
    - *Http Method*: il metodo HTTP che deve essere utilizzato per la chiamata al servizio di introspection;
    - *Posizione Token*: il metodo di passaggio del token al servizio di introspection. Sono supportati i classici metodi: HTTP Authorization Bearer, Header HTTP, Parametro URL e Parametro Form-Encoded Body. Negli ultimi tre casi sarà necessario fornire il nome dell'header o del parametro.
    - *Formato Risposta - Tipo*: indica il formato atteso della risposta. Maggiori dettagli sul mapping vengono forniti in "*Formati dei token*". I valori possibili sono:
        - *RFC 7662 - Introspection*: claims attesi definiti nel RFC "https://datatracker.ietf.org/doc/html/rfc7662#section-2.2";
        - *Google - TokenInfo*: claims definiti in "https://developers.google.com/identity/sign-in/web/backend-auth#calling-the-tokeninfo-endpoint";
        - *Personalizzato*: consente di definire un mapping puntuale tra il nome di un claim e l'informazione che GovWay cerca di estrarre dal token (Fig. 8.48);
        - *Plugin*: consente di indicare il nome di una classe che implementa una logica di parsing personalizzata (deve implementare l'interfaccia "org.openspcoop2.pdd.core.token.parser.ITokenParser").

OIDC - UserInfo

Sezione per attivare la richiesta al servizio di UserInfo per ottenere i dati inerenti l'utente possessore del token ricevuto (Fig. 8.47).

Per il corretto puntamento al servizio di UserInfo devono essere forniti in prima istanza i parametri generali legati all'endpoint riferito, che sono in comune con quelli del servizio di Token Introspection, e quindi già descritti in precedenza.

Successivamente si dovranno fornire i dati di configurazione specifici per il servizio UserInfo, che sono:

- *Tipo*: Si seleziona il tipo di servizio UserInfo riferito. I valori possibili sono:
    - *OpenID Connect - UserInfo*: servizio di UserInfo conforme allo standard OpenID Connect "https://openid.net/specs/openid-connect-core-1_0.html#UserInfo";

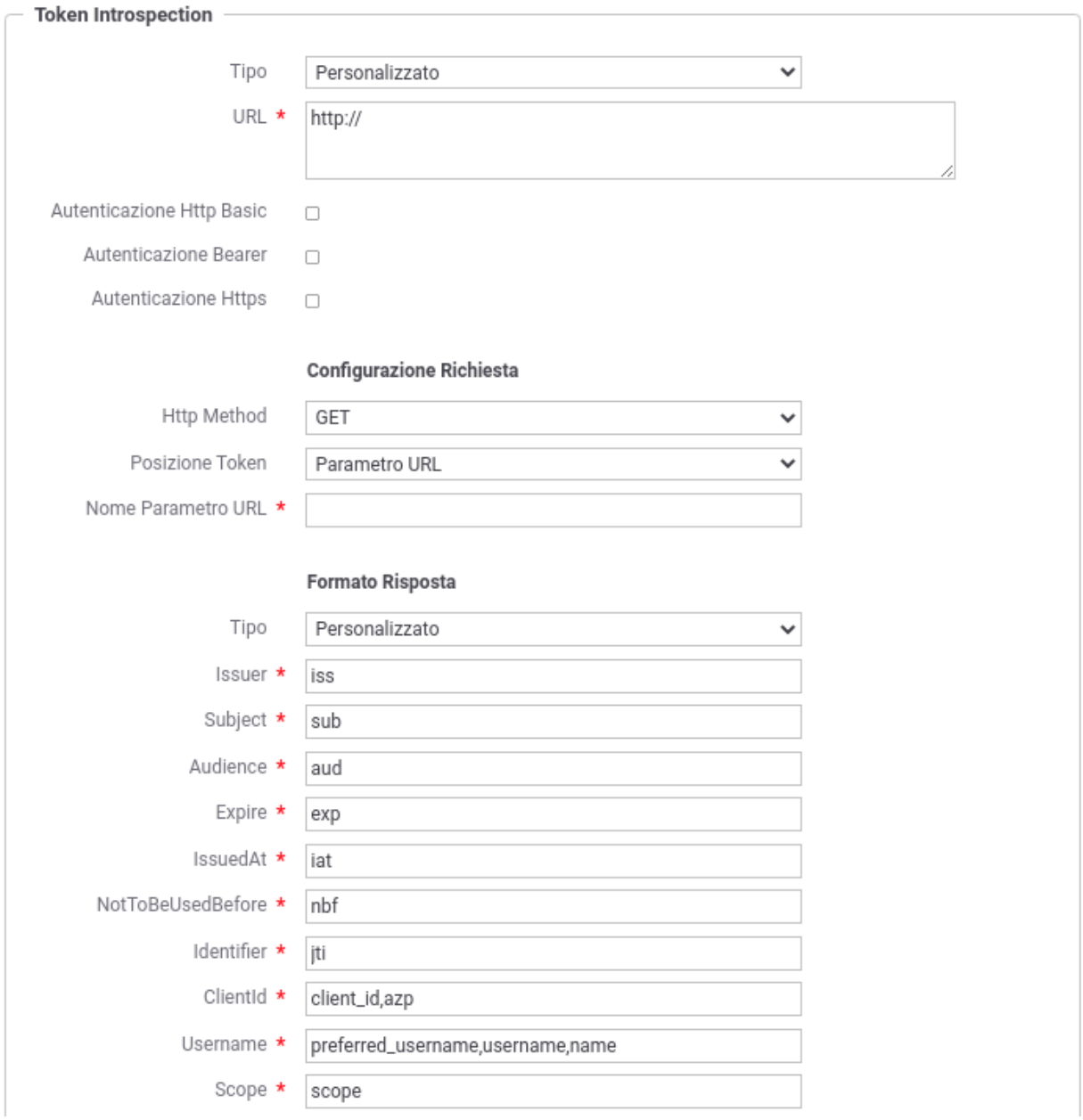


Fig. 8.46: Configurazione personalizzata del servizio di Token Instrospection

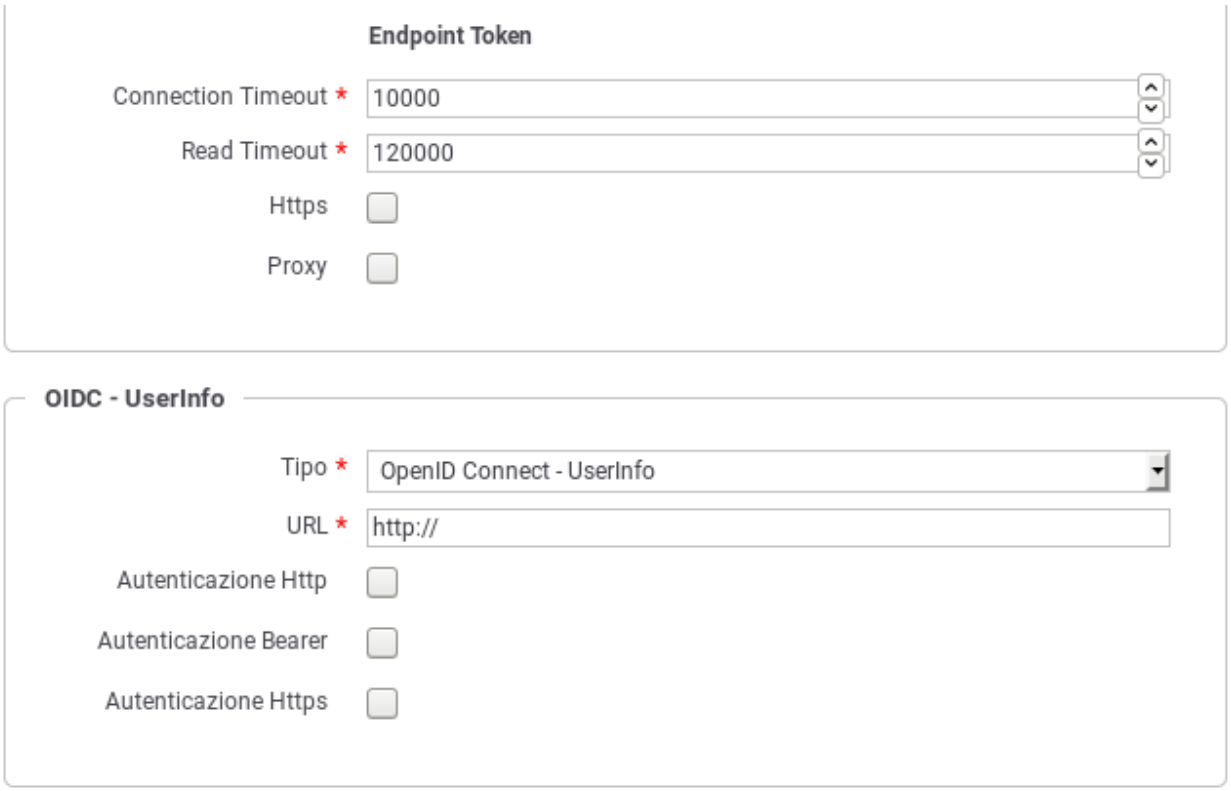


Fig. 8.47: Dati di puntamento al servizio di UserInfo

  - *Google - UserInfo*: servizio UserInfo di Google. La URL di default del servizio viene inserita automaticamente.
  - *Personalizzato*: si consente di fornire i dati di configurazione di un servizio personalizzato di UserInfo. I dati di configurazione sono gli stessi già descritti nel caso della configurazione del servizio di Token Introspection personalizzato.
- *URL*: La URL del servizio di UserInfo.
- *Autenticazione*: La configurazione del metodo di autenticazione, quando applicabile.

Token Forward

Azione di elaborazione che consiste nell'inoltro del token ricevuto al destinatario. Una volta attivata questa opzione, devono essere indicate le seguenti informazioni:

- *Originale*: opzione che consente di inoltrare il token originale al destinatario. Attivando questo flag è necessario specificare la modalità di inoltro a scelta tra le seguenti opzioni:
  - *Come è stato ricevuto*: Il token viene inoltrato al destinatario utilizzando lo stesso metodo con cui è stato ricevuto dal gateway.
  - *RFC 6750 - Bearer Token Usage (Authorization Request Header Field)*: Il token viene inoltrato al destinatario utilizzando l'header Authorization presente nella richiesta HTTP.
  - *RFC 6750 - Bearer Token Usage (URI Query Parameter)*: Il token viene inoltrato al destinatario tramite parametro access_token della Query String.

  - *Header HTTP*: Il token viene inoltrato al destinatario utilizzando un header HTTP il cui nome deve essere specificato nel campo seguente.
  - *Parametro URL*: Il token viene inoltrato al destinatario utilizzando un parametro della Query String il cui nome deve essere specificato nel campo seguente.

- *Informazioni Raccolte*: opzione disponibile quando è stata abilitata una delle azioni di validazione del token (introspection, user info o validazione JWT), consente di veicolare i dati ottenuti dal servizio di validazione, al destinatario. Una volta attivato il flag è necessario specificare la modalità di inoltro dei dati selezionando una tra le opzioni seguenti:
  - *GovWay Headers*: I dati raccolti dal token vengono inseriti nei seguenti header HTTP:

```
GovWay-Token-Issuer
GovWay-Token-Subject
GovWay-Token-Username
GovWay-Token-Audience
GovWay-Token-ClientId
GovWay-Token-IssuedAt
GovWay-Token-Expire
GovWay-Token-NotToBeUsedBefore
GovWay-Token-Scopes
GovWay-Token-FullName
GovWay-Token-FirstName
GovWay-Token-MiddleName
GovWay-Token-FamilyName
GovWay-Token-EMail
```

  - *GovWay JSON*: I dati raccolti dal token vengono inseriti in un oggetto JSON, il cui JsonSchema è il seguente:

```
{
    "required" : [ "id" ],
    "properties": {
        "id": {"type": "string"},
        "issuer": {"type": "string"},
    "subject": {"type": "string"},
    "username": {"type": "string"},
    "audience": {"type": "string"},
    "clientId": {"type": "string"},
    "iat": {
            "type": "string",
            "format": "date-time"
        },
    "expire": {
            "type": "string",
            "format": "date-time"
        },
    "expire": {
            "type": "string",
            "format": "date-time"
        },
        "roles": {
            "type": "array",
            "items": {"type": "string"}
        },
        "scope": {
            "type": "array",
            "items": {"type": "string"}
```

(continues on next page)

(continua dalla pagina precedente)

```
        },
    "userInfo": {
        "type": "object",
        "properties": {
            "fullName": {"type": "string"},
            "firstName": {"type": "string"},
            "middleName": {"type": "string"},
            "familyName": {"type": "string"},
            "email": {"type": "string"},
        },
        "additionalProperties": false
    }
    },
    "additionalProperties": false
}
```

Il JSON risultante viene inserito nell'Header HTTP *GovWay-Token*.

– *GovWay JWS*: I dati raccolti dal token vengono inseriti in un oggetto JSON, come descritto al punto precedente. In questo caso il token JSON viene inserito successivamente in un JWT e quindi firmato. Il JWS risultante viene inserito nell'Header HTTP *GovWay-JWT*.

– *JSON*: Le informazioni ottenute dai servizi di introspection, userinfo o il json estratto dal token jwt dopo la validazione, vengono inseriti negli header http o proprietà delle url indicati.

Nota: Le informazioni sono esattamente quelle recuperate dai servizi originali (o presenti nel token originale nel caso di validazione jwt).

- *JWS/JWE*: Uguale alla modalità JSON con la differenza che negli header http, o nelle proprietà delle url, vengono inseriti dei JWT firmati (caso JWS) o cifrati (caso JWE) contenenti al loro interno il JSON.

Formati dei token

Nella funzionalità "*Token Policy Validazione*" viene attuato un parsing del token ricevuto nel caso sia abilitata la "*Validazione JWT*" per estrarre le informazioni principali che vengono registrate da GovWay e possono essere inoltrate al backend sotto forma di header di integrazione ("*Token Forward*"). Un parsing delle informazione avviene inoltre anche se risulta attivata la funzionalità "*Token Introspection*" e/o "*OIDC - UserInfo*". Ogni funzionalità precedentemente indicata richiede che venga indicato il formato del token per poter interpretare correttamente le informazioni presenti. Di seguito viene fornita una tabella di mapping tra le informazioni che GovWay cerca di estrarre dal token e i nomi dei claims rispetto al formato impostabile nelle funzionalità suddette.

Tabella 8.1: Mapping informazione-claim per ogni formato di token

Informazione	RFC 7519 - JSON Web Token	RFC 7662 - Introspection	OpenID Connect - ID Token	Google - ID Token
Issuer	iss	iss	iss	iss
Subject	sub	sub	sub	sub
Audience	aud	aud	aud	aud
Expire	exp	exp	exp	exp
IssuedAt	iat	iat	iat	iat
NotToBeUsedBefore	nbf	nbf	non supportato	non supportato
Identifier	jti	jti	non supportato	non supportato
Scope	scope	scope	scope	scope
ClientId	non supportato	client_id	azp	azp
Username	non supportato	username	preferred_username o name	name
User Full name	non supportato	non supportato	name	name
User First name	non supportato	non supportato	given_name	given_name
User Middle name	non supportato	non supportato	middle_name	middle_name
User Family name	non supportato	non supportato	family_name o last_name	family_name
User eMail	non supportato	non supportato	email	email

All'interno di ogni funzionalità presente in "*Validazione JWT*" è anche inoltre possibile indicare un formato personalizzato che consente di definire un mapping puntuale tra il nome di un claim e l'informazione che GovWay cerca di estrarre dal token. Per ogni campo possono essere indicati più claims, separandoli con la virgola, ed in tal caso nel token verranno cercati nell'ordine in cui sono definiti. L'indicazione di un claim per ogni informazione non è vincolante rispetto alla presenza di tale claim all'interno del token.

8.6 Attribute Authority

Le Attribute Authority (AA) sono regolate dalle «Linee guida dei gestori di attributi qualificati» rilasciate da AGID ed operano erogando API che gestiscono gli attributi qualificati di persone fisiche o giuridiche.

Un attributo qualificato descrive una proprietà di un'identità e si definisce qualificato perché è attestato da un soggetto (Attribute Authority) cui la legge conferisce tale potere. La descrizione e il formato di ogni attributo è specifico dell'Attribute Authority alla quale è possibile richiedere attributi solo mediante la stipula di una convenzione. Inoltre le singole AA definiscono nelle proprie specifiche di integrazione quali siano gli elementi obbligatori che devono essere presenti nelle richieste tra cui l'informazione necessaria ad identificare il soggetto per cui si stanno richiedendo gli attributi.

GovWay supporta l'interazione con le Attribute Authority nella fase di verifica dell'autorizzazione all'accesso ad una API, permettendo di utilizzare gli attributi ottenuti dalle AA nelle politiche di accesso alle API.

Per poter definire politiche di controllo degli accessi basate sugli attributi è necessario registrare una o più AA. Queste potranno poi essere riferite nella configurazione delle singole API.

La gestione delle AA si effettua dalla sezione *Configurazione* > *Attribute Authority* della govwayConsole. Per registrarne una nuova si utilizza il pulsante *Aggiungi*. Il form di creazione appare inizialmente come quello illustrato in Fig. 8.49.

Inizialmente si inseriscono i dati identificativi:

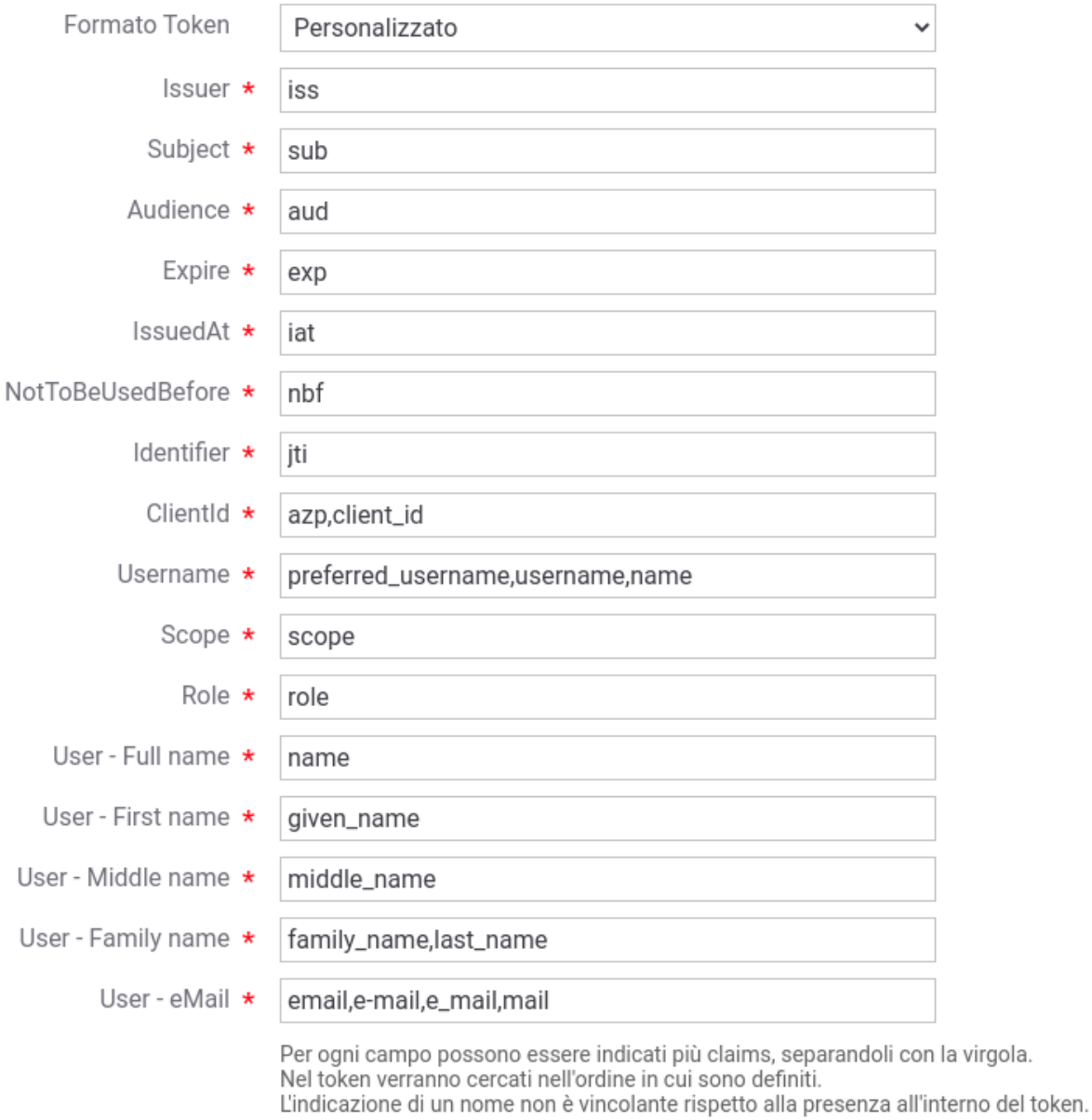


Fig. 8.48: Personalizzazione del formato del token

Fig. 8.49: Registrazione di una Attribute Authority

- *Nome*: nome univoco da assegnare all'AA
- *Descrizione*: testo di descrizione generale

Le sezioni successive dettagliano i criteri con cui si compone una richiesta di attributi e l'endpoint a cui deve essere spedita. Infine deve essere istruito GovWay su come interpretare la risposta di attributi ricevuta dall'AA.

8.6.1 Endpoint di una Attribute Authority

In questa sezione vengono descritti i parametri di connessione alla AA.

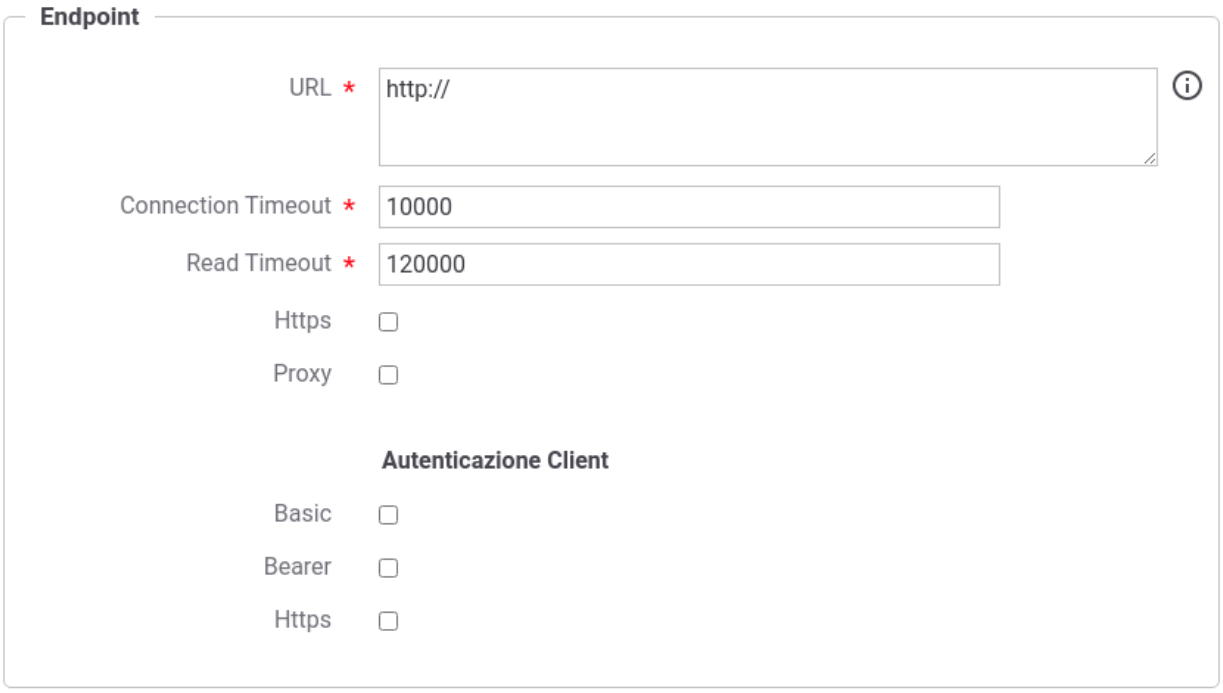


Fig. 8.50: Endpoint di un Attribute Authority

- *URL*: endpoint dell'AA a cui è possibile inviare una richiesta di attributi. Il valore può essere definito come costante o contenere parti dinamiche risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*).
- *Connection Timeout*: Tempo massimo in millisecondi di attesa per stabilire una connessione con il server.
- *Read Timeout*: Tempo massimo in millisecondi di attesa per la ricezione di una risposta dal server.
- *Https*: Parametri di configurazione nel caso in cui l'AA richieda un accesso Https.
- *Proxy*: Parametri di configurazione nel caso in cui l'AA richieda l'uso di un proxy per l'accesso.

Successivamente devono essere forniti i dati di configurazione specifici dell'autenticazione client, se richiesto dall'AA:

- *Autenticazione Http Basic*: flag da attivare nel caso in cui l'AA richieda autenticazione di tipo HTTP-BASIC. In questo caso dovranno essere forniti Username e Passwornd nei campi successivi.
- *Autenticazione Bearer*: flag da attivare nel caso in cui l'AA richieda autenticazione tramite un bearer token. Il token dovrà essere indicato nel campo successivo fornito.
- *Autenticazione Https*: flag da attivare nel caso in cui l'AA richieda autenticazione di tipo Https. In questo caso dovranno essere forniti tutti i dati di configurazione nei campi presenti nella sezione "https".

Nel caso sia attivato il flag relativo ad un Proxy o una configurazione Https saranno presentate delle sezioni omonime dove poter inserire i dati di configurazione richiesti.

8.6.2 Richiesta di Attributi

Ogni singola AA definisce nella propria interfaccia quali siano gli elementi obbligatori che devono essere presenti nelle richieste, tra cui l'informazione necessaria ad identificare il soggetto a cui si riferiscono gli attributi richiesti.

La sezione seguente consente di definire come GovWay debba formare la richiesta che verrà inoltrata all'endpoint configurato nella sezione *Endpoint di una Attribute Authority*.

- *Posizione*: indica dove risiede la richiesta di attributi nella comunicazione HTTP:
  - *Authorization Bearer*: richiesta inserita nell'header HTTP "Authorization" con prefisso "Bearer";
  - *HTTP Payload*: richiesta veicolata come payload http;
  - *Header HTTP*: richiesta inserita in un header HTTP il cui nome viene definito nel campo successivo fornito;
  - *Parametro URL*: richiesta inserita come parametro della url il cui nome viene definito nel campo successivo fornito.
- *Http Method*: consente di selezionare il tipo di richiesta HTTP da utilizzare tra quelle compatibili con la *Posizione* della richiesta di attributi.
- *Tipo Richiesta*: indica il formato della richiesta:
  - *JWS*: token JWT firmato (https://datatracker.ietf.org/doc/html/rfc7515);
  - *JSON*: consente di definire la richiesta di attributi tramite la definizione di un template;
  - *Personalizzata*: simile alla precedente opzione, consente inoltre di impostare il Content-Type associato alla richiesta.

Richiesta nel formato JWT

Nel caso di richiesta di tipo *JWS* si devono fornire le informazioni necessarie a produrre il JWT firmato così suddivise:

- *JWS KeyStore*: dati di accesso al keystore contenente la chiave privata ed il certificato da utilizzare per firmare il JWT.

- *JWS Header*: consente di indicare quali dati debbano essere inseriti nella parte header (non firmati) del JWT; tra i parametri impostabili vi sono l'algoritmo di firma e l'indicazione se deve essere inserito il certificato utilizzato per la firma nell'header (x5c).
- *JWS Payload*: consente di impostare i valori dei claim presenti nella parte body (firmata) del JWT. Vengono fornite differenti modalità con le quali poter definire il payload:
  - *RFC7515*: consente di definire i claims standard ("iss","sub" e "aud") e la validità temporale del JWT. Inoltre è possibile definire ulteriori claims da inserire nel body indicandoli per riga (nome=valore) nel campo "Claims". I claim "iss", "sub", "aud" e gli eventuali claims aggiuntivi possono essere definiti tramite costanti o possono contenere parti dinamiche risolte a runtime dal Gateway (per maggiori dettagli *Valori dinamici*).
  - *Template*: il payload viene definito tramite un template che può contenere parti dinamiche risolte a runtime definite tramite una sintassi proprietaria di GovWay.
  - *Freemarker Template*: il payload viene definito utilizzando il template «Freemarker» (https://freemarker.apache.org/).
  - *Velocity Template*: il payload viene definito utilizzando il template «Velocity» (http://velocity.apache.org/).

Richiesta in altri formati

Nel caso di richiesta di tipo *Json* o *Personalizzata* si deve fornire un template che definisce la richiesta di attributi. Il tipo di template utilizzabile è selezionabile tra i seguenti:

- *Template*: il contenuto della richiesta viene definito tramite un template che può contenere parti dinamiche risolte a runtime definite tramite una sintassi proprietaria di GovWay;
- *Freemarker Template*: il contenuto della richiesta viene definito utilizzando il template «Freemarker» (https://freemarker.apache.org/);
- *Velocity Template*: il contenuto della richiesta viene definito utilizzando il template «Velocity» (http://velocity.apache.org/).

Valori dinamici utilizzabili nei Template

I costrutti utilizzabili nei template sono gli stessi utilizzabili per la funzionalità di trasformazione, come descritti nella sezione "*Valori dinamici*", arricchiti di un'ulteriore istruzione che consente di individuare gli attributi da richiedere, così come configurati sulla specifica fruizione o erogazione di API nella quale è stata riferita l'AA :

- *requiredAttributes:METHOD* : il valore "METHOD" fornito deve rappresentare un metodo valido all'interno della classe "org.openspcoop2.pdd.core.token.attribute_authority.RequiredAttributes"
  - Se la richiesta è definita tramite un template con la sintassi specifica di GovWay, gli attributi saranno direttamente accessibili utilizzando il formato "${requiredAttributes:METHOD}"; ad es. per ottenere la lista degli attributi in un formato utilizzabile all'interno di un array json usare ${requiredAttributes:jsonList()} oppure ${requiredAttributes:formatList(«,»)}.
  - Se la richiesta è definita tramite template Freemarker o Velocity, l'oggetto contenente gli attributi da richiedere è presente nel contesto con chiave di accesso "aa".

Di seguito un esempio di template GovWay che definisce una richiesta JSON in cui l'identità della persona fisica per cui si richiedono gli attributi viene prelevata dal token OAuth e gli attributi richiesti sono quelli configurati nell'erogazione di API:

```
{
        "attributes": [${requiredAttributes:jsonList()}],
        "fiscalCode": "${tokenInfo:username}"
}
```

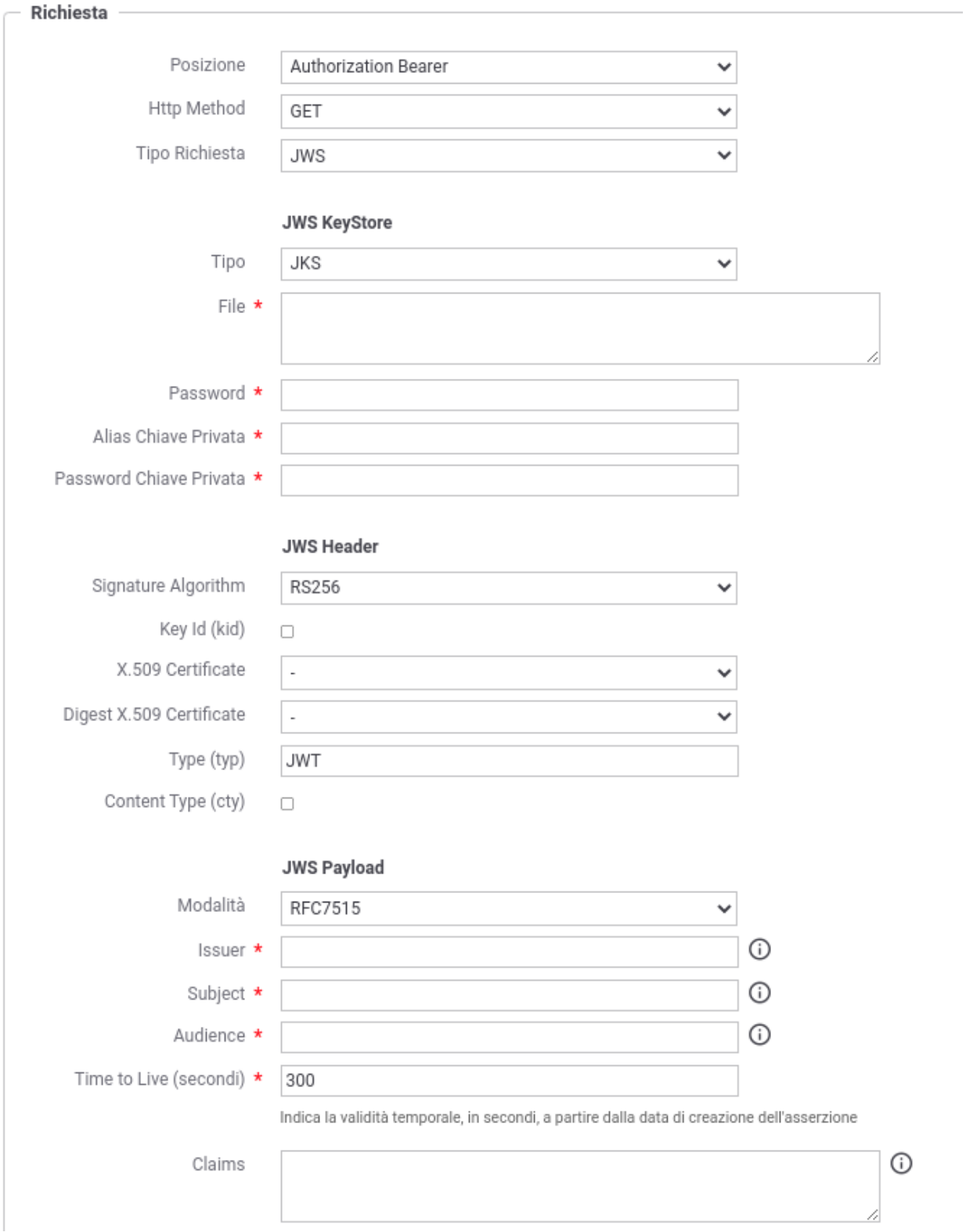


Fig. 8.51: Richiesta di Attributi nel formato JWS con modalità “RFC7515”

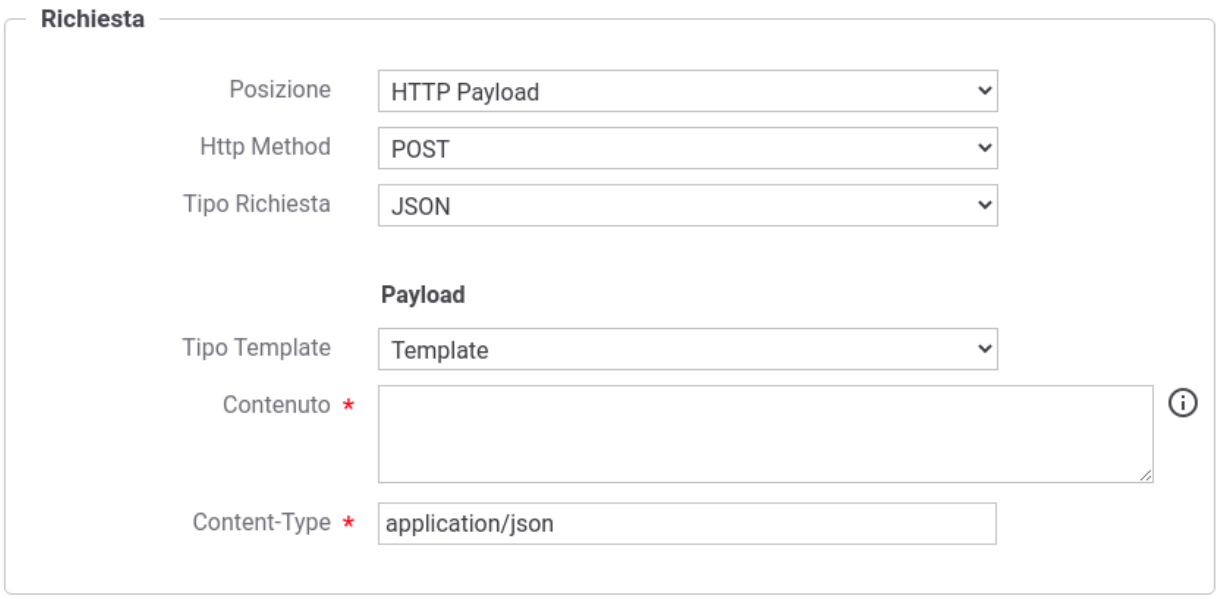


Fig. 8.52: Richiesta di Attributi nel formato JSON

8.6.3 Risposta della Attribute Authority

Ogni singola AA utilizza un proprio formato per la descrizione degli attributi nella risposta fornita. Il tipo della risposta deve essere definito nel campo "*Tipo Risposta*". Di seguito vengono descritte le opzioni richieste per ogni tipo.

- *JWS*: la risposta viene gestita come token JWT firmato (https://datatracker.ietf.org/doc/html/rfc7515) presente nel payload http. Deve essere indicato il claim che contiene gli attributi richiesti ed è possibile elencare più claim separandoli tramite virgola. Nella sezione "*TrustStore*" devono essere indicati i dati che consentono di accedere al truststore da utilizzare per validare il token jws.
- *JSON*: la risposta viene processata come messaggio JSON. Se gli attributi sono contenuti in uno o più elementi devono esserne elencati i nomi separandoli tramite virgola. Invece lasciando vuoto il campo "*Attributi*" tutti gli elementi presenti saranno interpretati come attributi.
- *Personalizzata*: la risposta viene processata tramite la classe indicata nel campo "*ClassName*". La classe fornita deve implementare l'interfaccia "org.openspcoop2.pdd.core.token.attribute_authority.IRetrieveAttributeAuthorityResponseParser".

8.7 Tags

La sezione *Configurazione* > *Tags* è dedicata alla gestione dei tags che possono essere utilizzati per la classificazione delle API presenti nel registro.

I tags possono essere creati direttamente durante la registrazione di una API, oppure da questa sezione in maniera più sistematica e assegnando loro un tipo, Soap o Rest, che indica l'ambito di utilizzo del tag stesso.

La sezione mostra l'elenco dei tags disponibili (Fig. 8.56).

L'elenco dei tag può essere filtrato impostando, nella barra dei filtri a comparsa, un pattern per il nome o un tipo. Oltre ad aggiungere ed eliminare i tag esistenti è possibile esportarli in blocco.

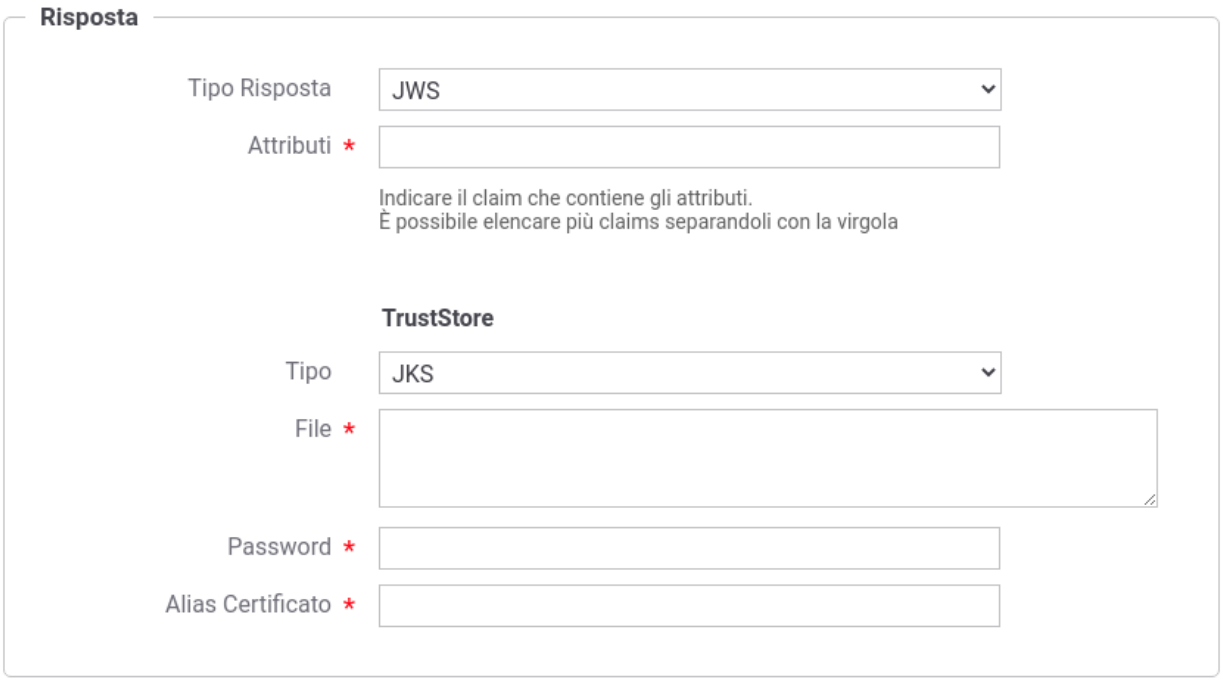


Fig. 8.53: Risposta di Attributi nel formato JWS

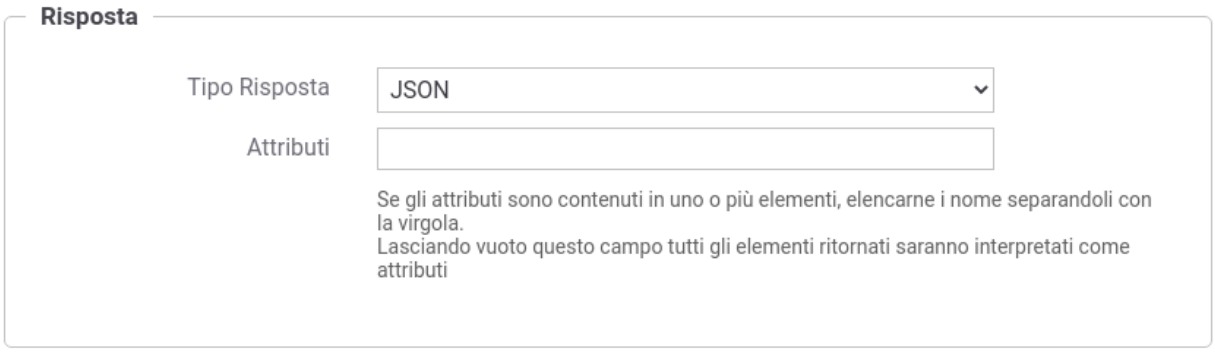


Fig. 8.54: Risposta di Attributi nel formato JSON

Fig. 8.55: Risposta di Attributi in un formato personalizzato

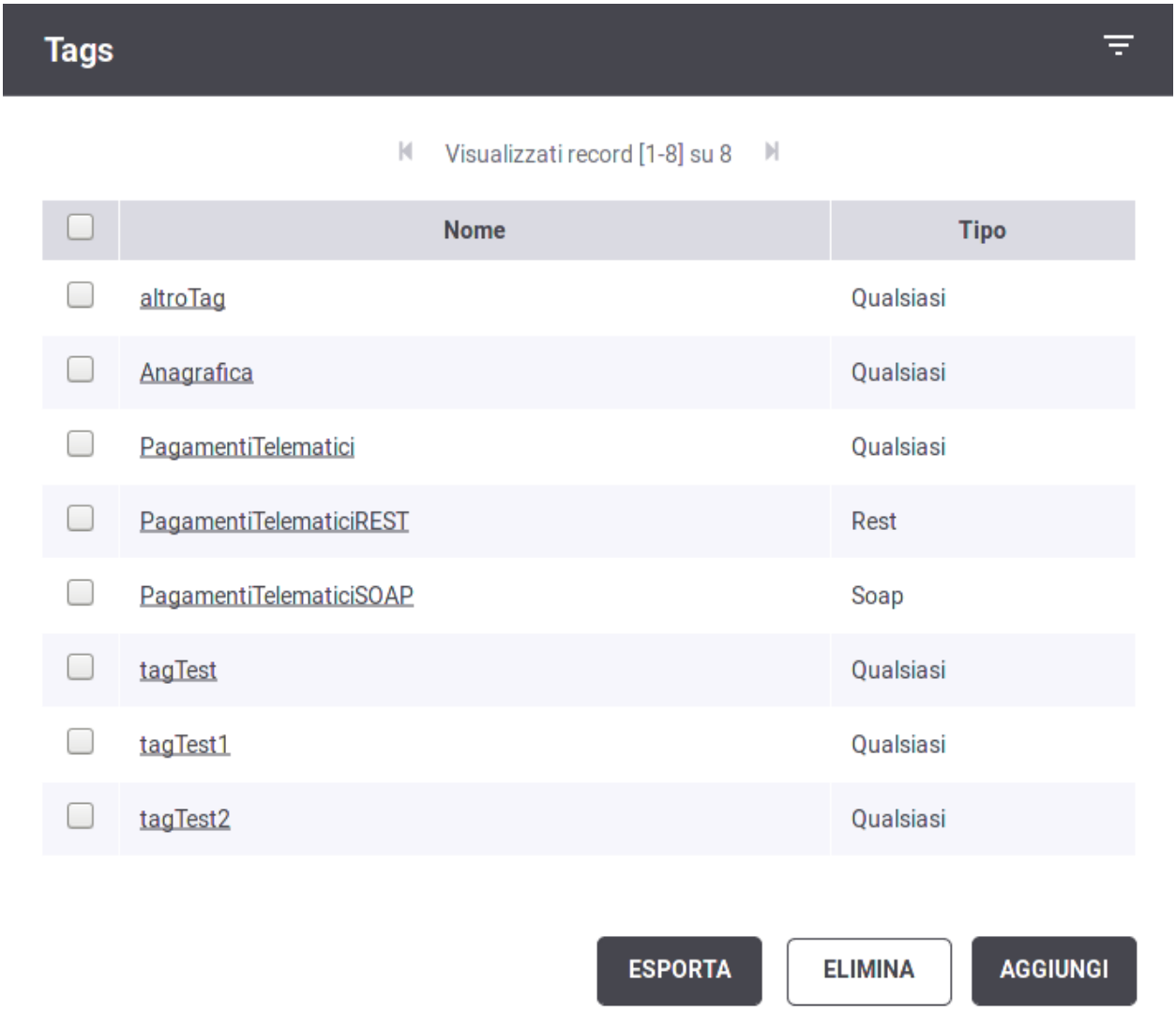


Fig. 8.56: Elenco dei tags

Col pulsante *Aggiungi* si apre il form per creare un nuovo tag (Fig. 8.57).

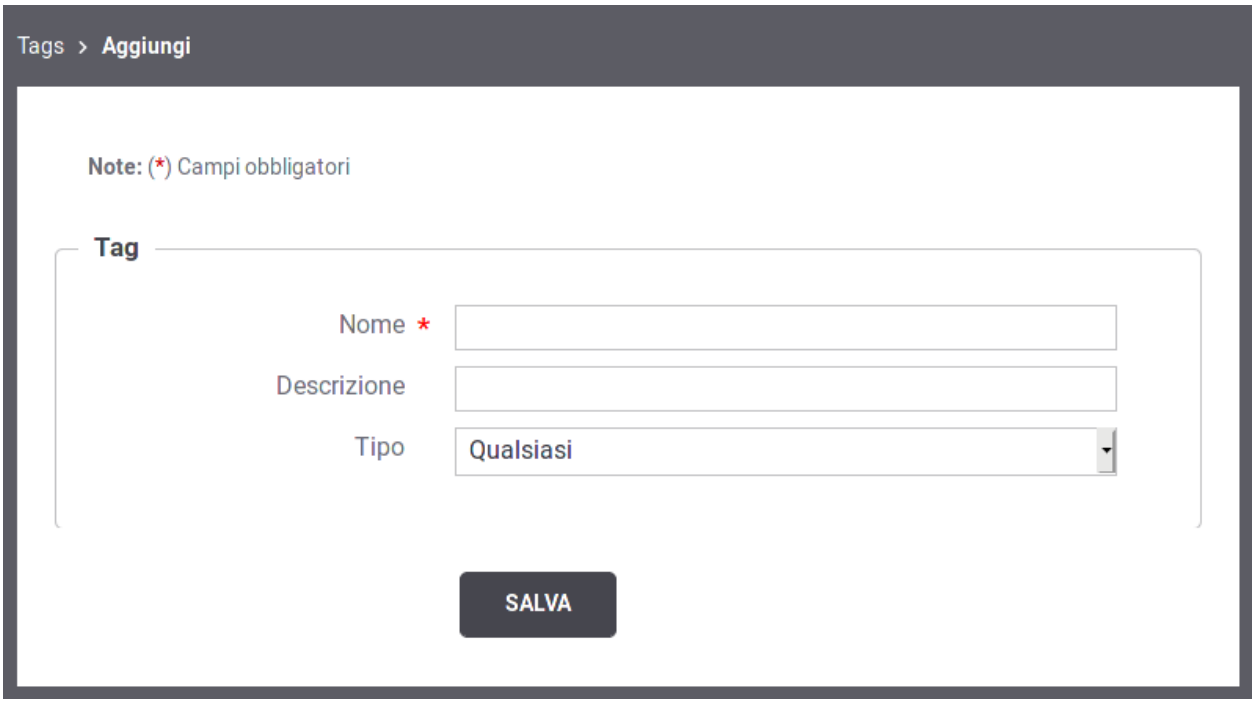


Fig. 8.57: Creazione di un tag

Per creare un tag si inseriscono i seguenti dati:

- *Nome*: il nome del tag
- *Descrizione*: descrizione del tag
- *Tipo*: serve per indicare per quali API è possibile utilizzare il tag: SOAP, REST o Qualsiasi.

8.8 Utenti

La sezione *Configurazione* > *Utenti* è dedicata alla gestione degli utenti dei cruscotti grafici govwayConsole e govwayMonitor.

Prima di descrivere le funzionalità relative alla gestione utenti è necessario fare una premessa sull'organizzazione dei permessi che sono assegnabili ad un utente.

Le funzionalità delle console grafiche sono partizionate in gruppi cui corrispondono puntuali permessi che possono essere concessi agli utenti per limitarne l'operatività. Vediamo quali sono i gruppi funzionali, e conseguentemente i permessi associabili a ciascun utente:

- *Registro*
  - *Gestione API [S]* - Gestione delle entità di configurazione dei servizi, quali: API, Erogazioni, Fruizioni, ecc.
- *GovWay Monitor*
  - *Monitoraggio [D]* - Accesso alle funzionalità di monitoraggio della console govwayMonitor.

  - *Reportistica [R]* - Accesso alle funzionalità di reportistica della console govwayMonitor.
- *Strumenti*
  - *Auditing [A]* - Accesso alle funzionalità di consultazione delle tracce del servizio di Auditing.
- *Configurazione*
  - *[C]* - Accesso alle funzionalità di configurazione. Queste funzionalità sono quelle presenti nel menu di navigazione nel gruppo *Configurazione* e riguardano: tracciamento, controllo del traffico, import-export, ecc.
  - *[U]* - Possibilità di gestire gli utenti delle console. Gli utenti con questo permesso, sono di fatto dei superutenti in quanto possono assumere l'identità di un qualunque utente del sistema.
- *Altri Permessi (visibili solo configurazione specifica del prodotto)*
  - *[P]* - Gestione delle entità di configurazione degli Accordi di Cooperazione e Servizi Composti.
  - *[M]* - Accesso alle code messaggi sul gateway. Questa autorizzazione consente ad esempio di consultare i messaggi presenti nelle Message Box dell'Integration Manager ed eventualmente effettuare delle rimozioni.

L'applicazione, al termine dell'installazione, contiene una utenza (credenziali indicate durante l'esecuzione dell'installer) che permette di effettuare tutte le principali operazioni di gestione.

Gli utenti in possesso del permesso [U] possono creare dei nuovi utenti. La maschera di creazione di un nuovo utente è quella mostrata in Fig. 8.58.

Le informazioni da inserire sono:

- *Informazioni Utente*
  - *Nome*
- *Permessi di Gestione*: sezione che consente di assegnare i permessi all'utente e quindi decidere quali funzionalità rendergli accessibili.
- *Profilo di Interoperabilità*: sezione che consente di decidere quali, tra i profili disponibili, rendere accessibili all'utente.
- *Visibilità dati tramite govwayMonitor*: questa sezione è visibile solo se è stato abilitato uno dei permessi «GovWay Monitor». In questo contesto è possibile stabilire la visibilità dell'utente sulla console GovWay Monitor riguardo i seguenti:
  - *Soggetti*: opzione visibile solo se attiva la modalità multi-tenant, consente di limitare la visibilità delle entità di monitoraggio ai soli soggetti interni indicati in una whitelist. Per configurare la whitelist è necessario salvare l'utente da creare e successivamente accedere in editing. In alternativa è possibile attivare il flag «Tutti» per non assegnare limitazioni.
  - *API*: consente di limitare la visibilità delle entità di monitoraggio alle sole API indicate in una whitelist. Per configurare la whitelist è necessario salvare l'utente da creare e successivamente accedere in editing. In alternativa è possibile attivare il flag «Tutti» per non assegnare limitazioni.
- *Modalità Interfaccia*: opzione per decidere quale modalità, tra standard e avanzata, è quella di default per l'utente.
- *Password*: sezione per l'impostazione della password dell'utente.

Nota: I criteri minimi di sicurezza che una password deve soddisfare sono configurabili agendo sul file <directory-lavoro>/consolePassword.properties:

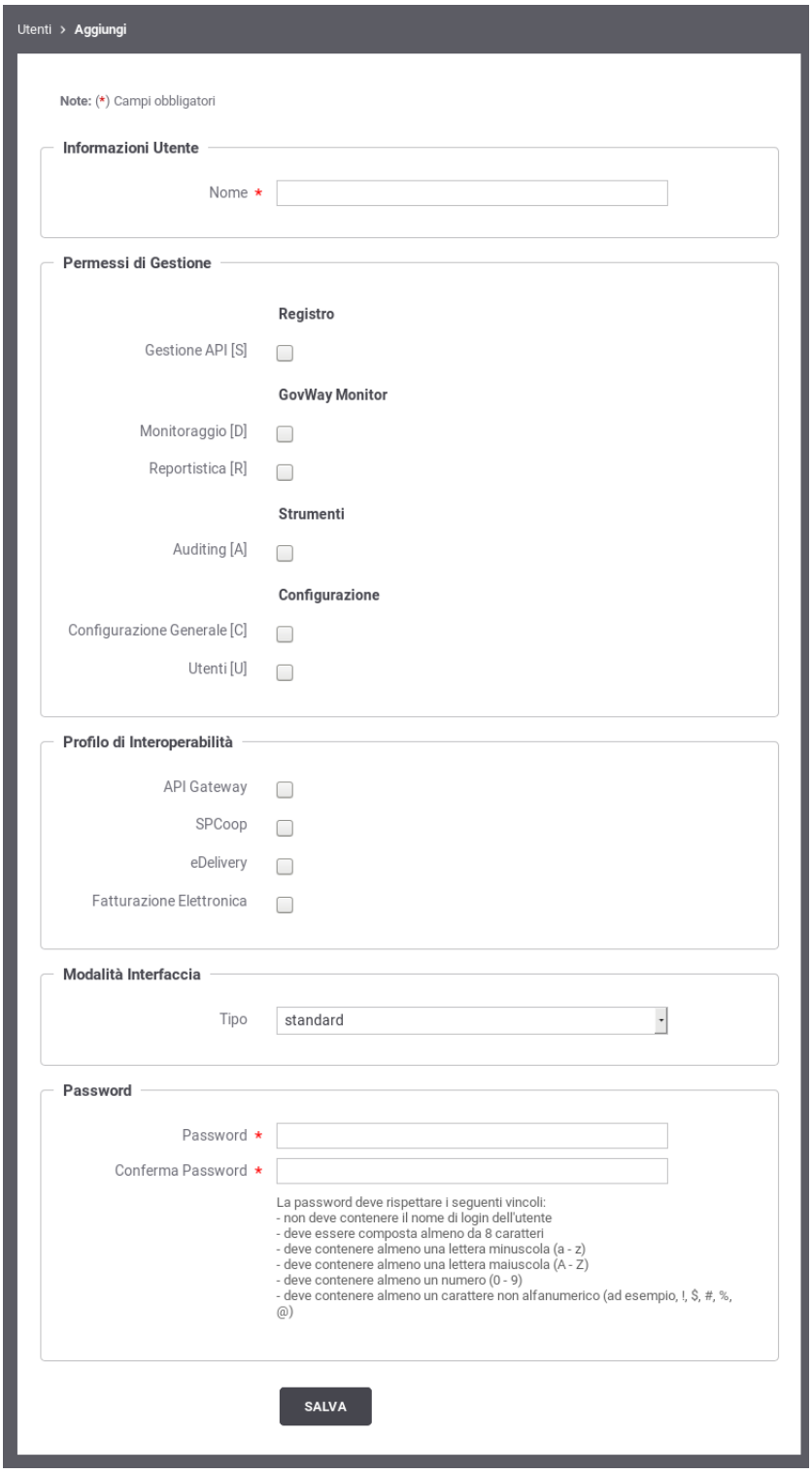


Fig. 8.58: Creazione nuovo utente

```
# Abilitare l'opzione seguente per poter autenticare:
# La password deve rispettare tutti i vincoli impostati

# Deve soddisfare le seguenti espressioni regolari
#passwordVerifier.regularExpression.EXP1=reg1
#..
#passwordVerifier.regularExpression.EXPN=regn

# Non deve contenere il nome di login dell'utente
passwordVerifier.notContainsLogin=true

# Non deve corrispondere ad una delle seguenti parole riservate
#passwordVerifier.restrictedWords=root, admin, administrator, amministratore

# Deve essere composta almeno da x caratteri
passwordVerifier.minLength=8

# Non deve essere composta da più di x caratteri
#passwordVerifier.maxLength=20

# Deve contenere almeno una lettera minuscola (a - z)
passwordVerifier.lowerCaseLetter=true

# Deve contenere almeno una lettera maiuscola (A - Z)
passwordVerifier.upperCaseLetter=true

# Deve contenere almeno un numero (0 - 9)
passwordVerifier.includeNumber=true

# Deve contenere almeno un carattere non alfabetico (ad esempio, !, $, #, %,␣
↪@)
passwordVerifier.includeNotAlphanumericSymbol=true

# Tutti i caratteri utilizzati devono essere differenti
#passwordVerifier.allDistinctCharacters=true

# La password dovrà essere aggiornata ogni 90 giorni
# Impostare un valore <=0 per disabilitare la verifica
#passwordVerifier.expireDays=90
passwordVerifier.expireDays=-1

# Abilita lo storico delle password non consentendo di aggiornare la␣
↪password corrente con una precedentemente già impostata.
passwordVerifier.history=true
```

La pagina indice della sezione Utenti visualizza gli utenti già presenti nel sistema con i relativi permessi e i link per modificarli o assumerne l'identità (Fig. 8.59)

Nota: La password generata e assegnata all'utente viene visualizzata solamente nell'avviso visualizzato in seguito alla creazione (Fig. 8.60) e successivamente non è più consultabile.

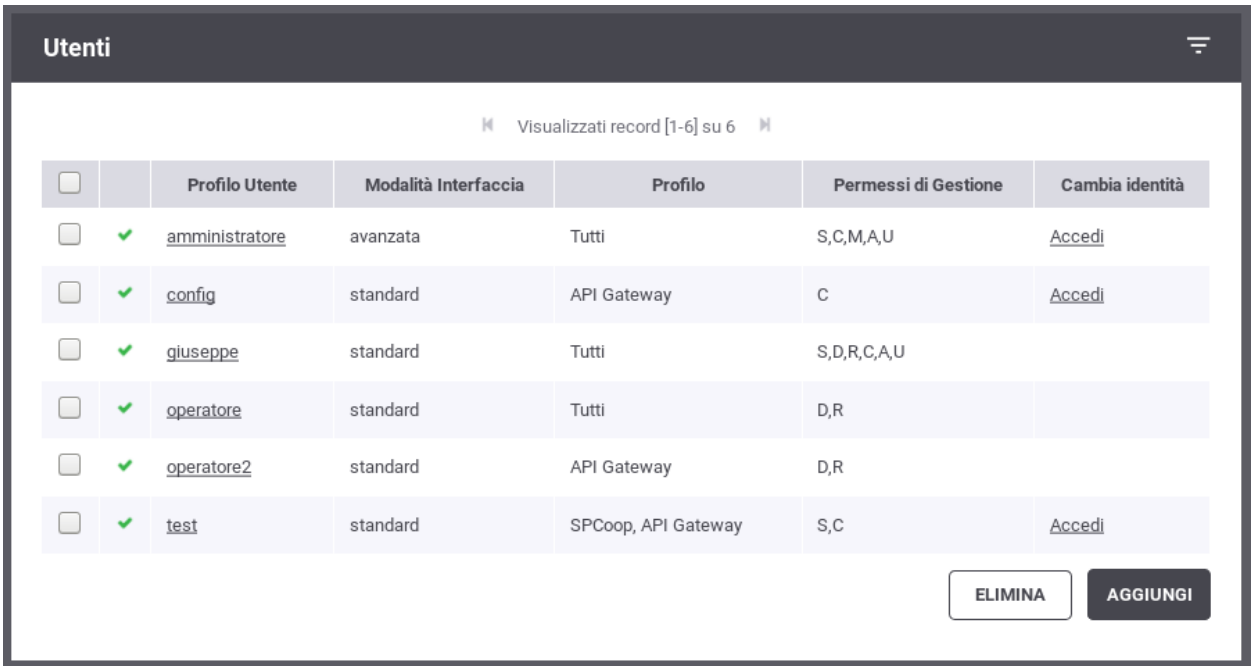


Fig. 8.59: Lista degli utenti

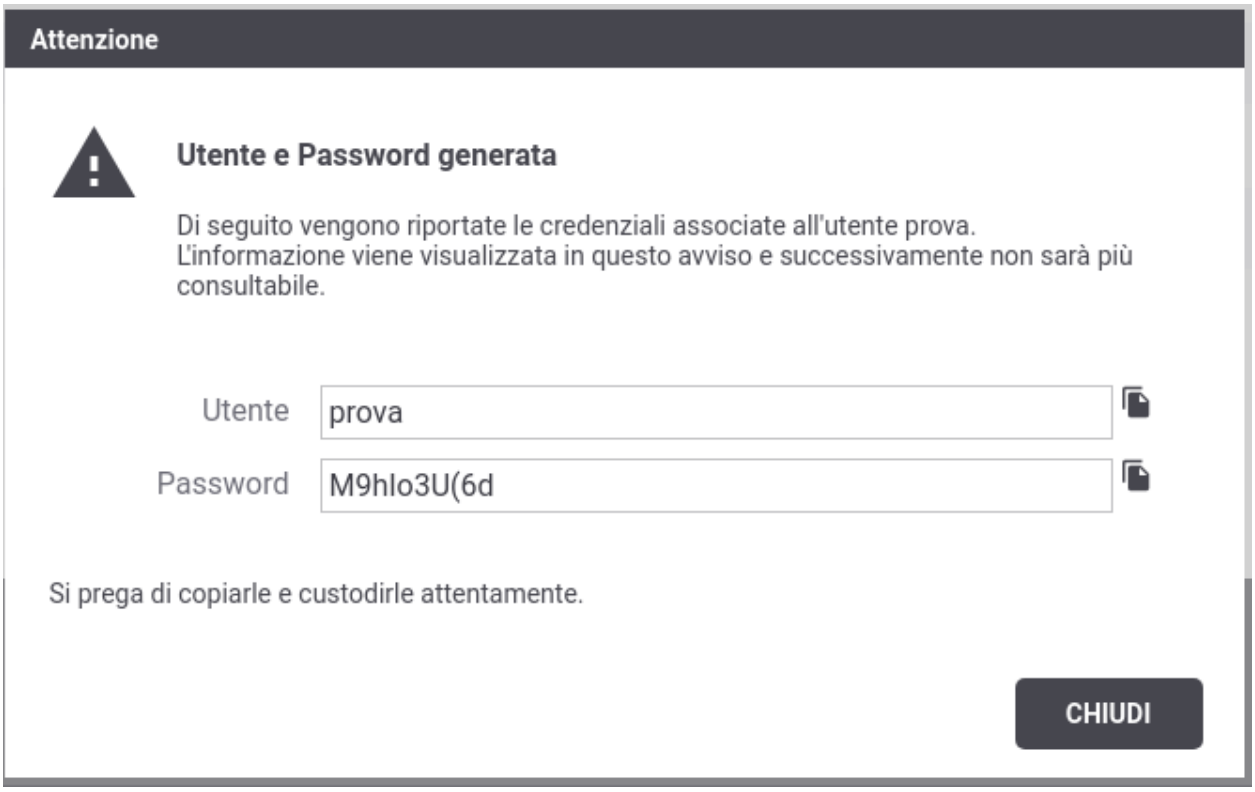


Fig. 8.60: Avviso di copia delle credenziali dell'utente

Nel caso di smarrimento della password è necessario procedere con la generazione di una nuova password (Fig. 8.61).

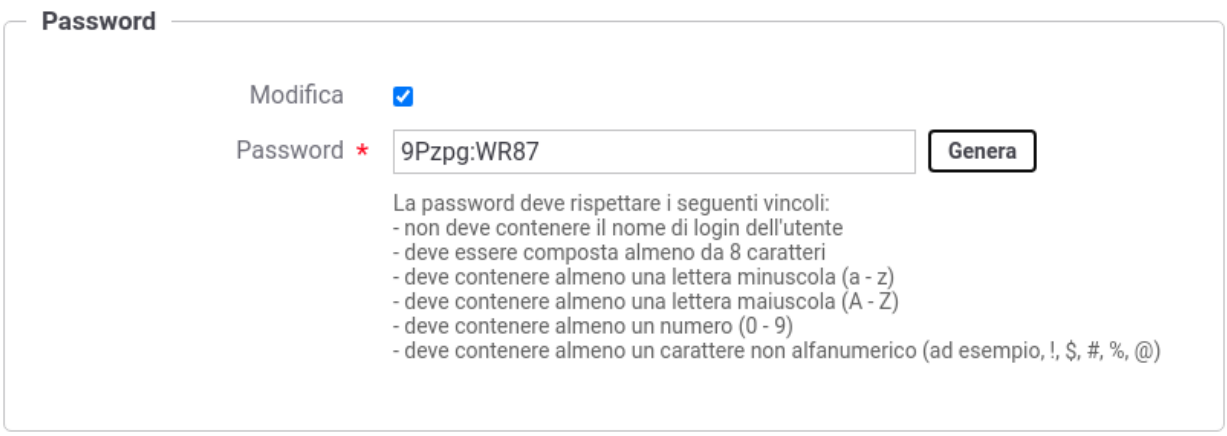


Fig. 8.61: Aggiornamento delle credenziali dell'utente

8.9 Importa

L'importazione di entità nel registro può essere effettuata tramite la sezione accessibile con la voce di menu *Importa* presente nella sezione *Configurazione*.

Gli archivi che possono essere importati devono essere nel formato atteso da govway e sono ottenibili:

- attraverso un'esportazione effettuabile tramite govwayConsole come indicato nella sezione *Esporta*
- scaricando le govlets disponibili sul sito del progetto che permettono di pre-configurare GovWay per una specifica API

Il form che compare per l'importazione è quello riportato in Fig. 8.62. I passi da eseguire sono i seguenti:

- *Validazione Documenti* (disponibile solamente con interfaccia in modalità avanzata, per default è abilitato): Se attivato, questo flag indica che i documenti presenti nell'archivio vengono validati prima di essere importati (es. wsdl, xsd, openapi 3, swagger 2 . . .).
- *Aggiornamento*: Se attivato, questo flag indica che l'archivio da importare costituisce un aggiornamento del registro attuale. Gli elementi presenti nell'archivio, che risultano già esistere sul registro di GovWay, verranno aggiornati solamente se il flag viene abilitato.
- *Policy di Configurazione*: eventuali policy globali (Token, Rate Limiting) presenti nell'archivio verranno importate solamente se il flag viene abilitato.
- *Configurazione di GovWay*: una eventuale configurazione presente nell'archivio verrà importata solamente se il flag viene abilitato.
- Selezionare dal filesystem il file che corrisponde all'archivio che deve essere importato.

Nota: Attraverso l'abilitazione di configurazioni avanzate relative ai Profili di Interoperabilità può comparire una ulteriore scelta iniziale che serve ad indicare la modalità cui fanno riferimento le entità contenute nell'archivio da importare.

Ad esempio, per il Profilo SPCoop se viene abilitata la proprietà "org.openspcoop2.protocol.spcoop.packageSICA" nel file locale "/etc/govway/spcoop_local.properties", verrà richiesto quale tipo di archivio si vuole importare a scelta tra:

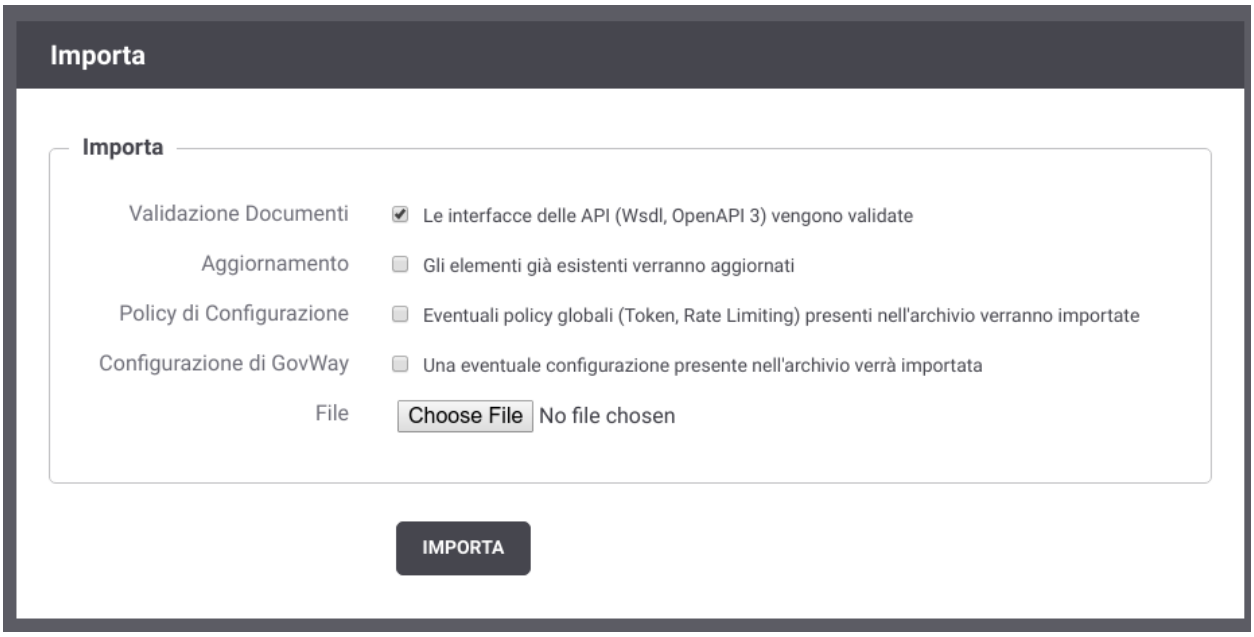


Fig. 8.62: Importazione di entità nel registro

- *spcoop*: il formato standard basato sulle specifiche SPCoop
- *govlet*: il formato di govway

8.10 Esporta

L'esportazione dei dati di configurazione dalla govwayConsole è possibile nei modi seguenti:

- Selezionando singolarmente le entità di configurazione da esportare, come ad esempio «Erogazioni» o «API», e premendo il pulsante *Esporta* (Fig. 8.63).

  Dopo aver selezionato il pulsante «Esporta», una seconda maschera (Fig. 8.64) riporta le seguenti informazioni:

  - *Profilo Interoperabilità*: indicazione del profilo cui fa riferimento l'esportazione.
  - *Soggetto*: indicazione del dominio cui fa riferimento l'esportazione.
  - *Tipologia archivio*: se previsto dal Profilo di Interoperabilità, fa selezionare la tipologia di archivio da produrre. Il default è il formato *Govlet* standard di esportazione di Govway.
  - *Policy di Configurazione*: se il flag viene abilitato vengono incluse nell'archivio esportato le policy globali (Token, Rate Limiting) condivise tra più API
  - *Elementi di Registro*: se il flag viene abilitato vengono incluse nell'archivio esportato anche gli elementi del registro riferiti da quelli selezionati.

- Tramite la voce di menu *Configurazione* > *Esporta* che presenta le opzioni mostrate in Fig. 8.65.

  Le opzioni presenti sono:

  - *Profilo Interoperabilità*: indica quale profilo riguarda l'esportazione che si sta effettuando
  - *Tipologia Archivio*: nei casi che lo prevedono, consente di specificare il formato dell'archivio di esportazione da produrre.

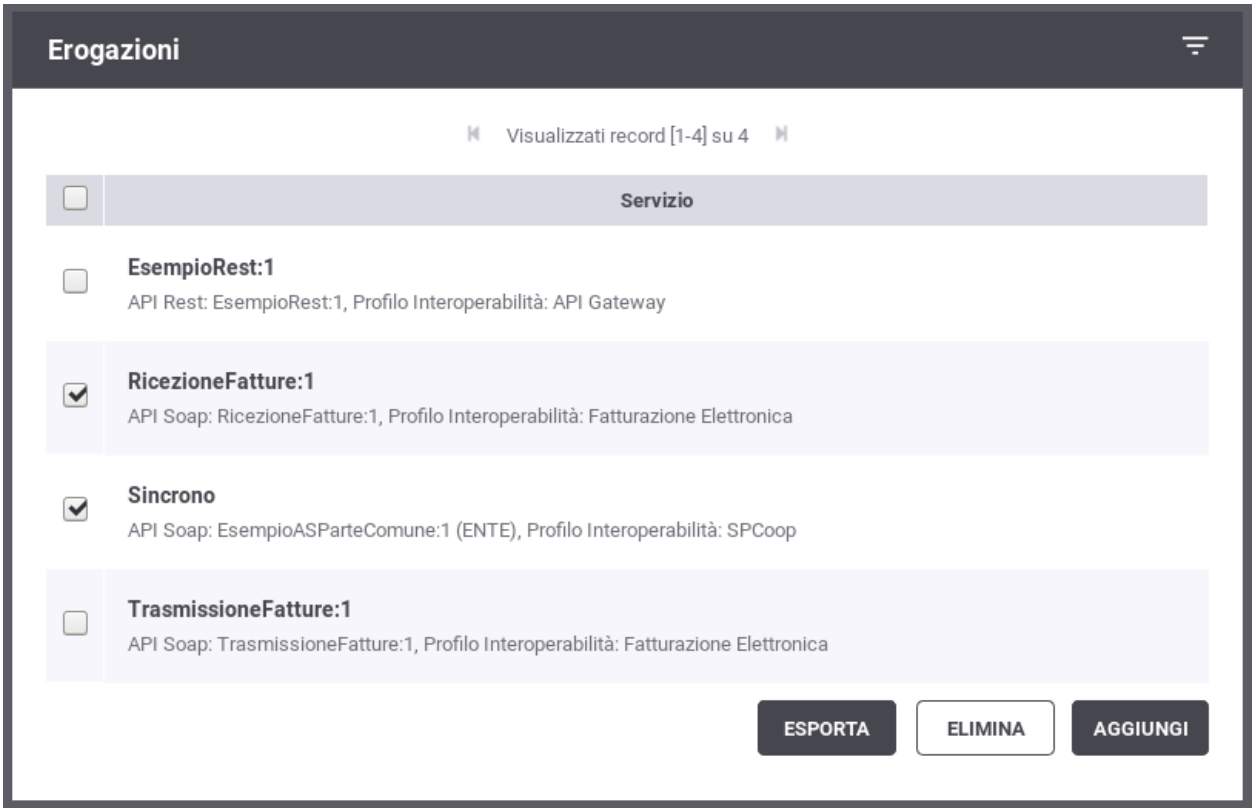


Fig. 8.63: Esportazione di singole entità del registro

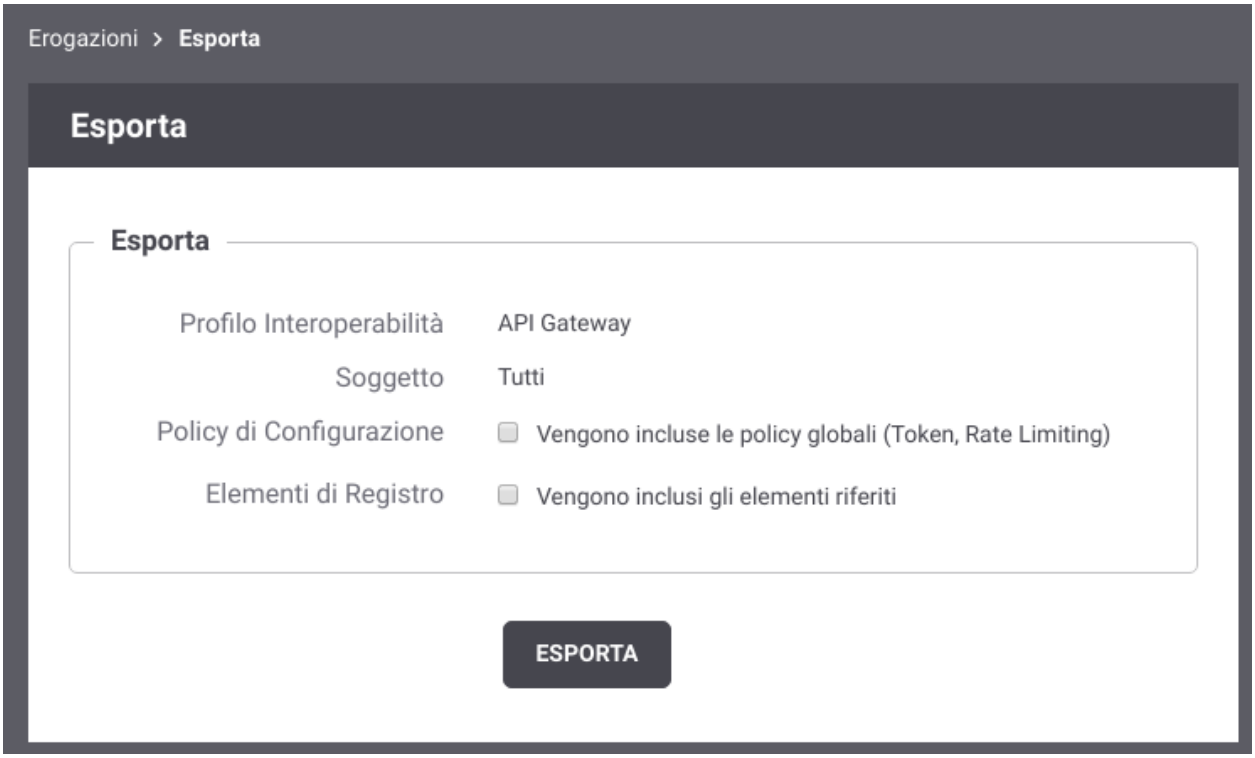


Fig. 8.64: Esportazione di entità nel registro: parametri

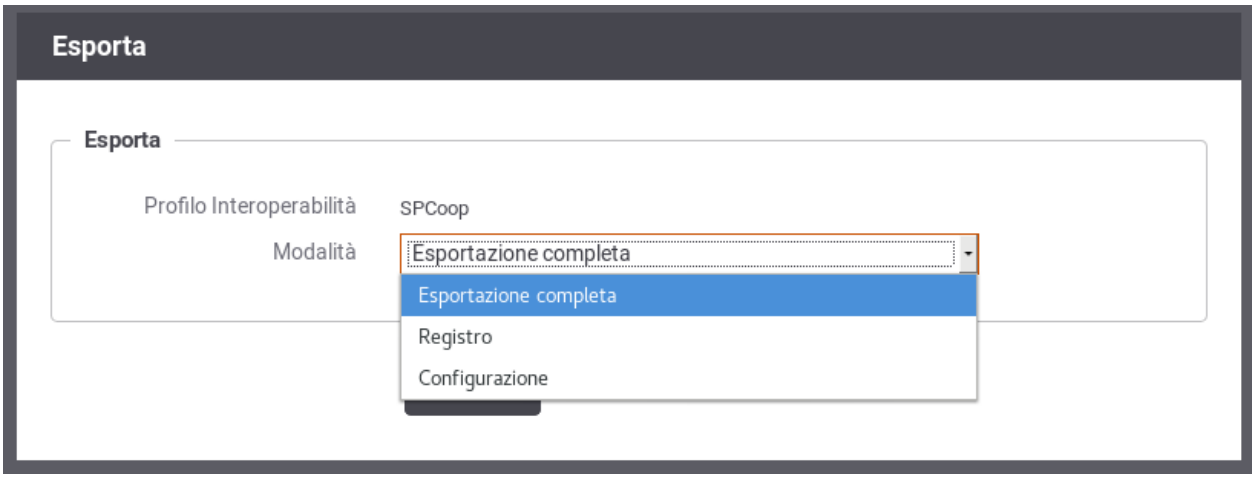


Fig. 8.65: Esportazione di entità nel registro

- *Modalità*: consente di specificare cosa esportare tra le seguenti possibilità:
  - *Esportazione completa*: esportazione dell'intero repository di configurazione (limitatamente al profilo di interoperabilità selezionato, se diverso da «Tutti»).
  - *Registro*: esporta solo le entità del registro (erogazioni, fruizioni, api, ecc)
  - *Configurazione*: esporta solo le entità della sezione Configurazione (token policy, tracciamento, ecc).

Il formato dell'archivio prodotto come risultato dell'esportazione dipende dalla modalità cui fanno riferimento le entità selezionate.

8.11 Auditing

In questa sezione descriviamo le modalità di configurazione del servizio di auditing, al fine di definire quali informazioni devono essere tracciate, con che formato e con che livello di dettaglio.

Gli utenti con permesso [C] Configurazione (vedi sezione *Utenti*) hanno la possibilità di configurare il servizio di auditing, al fine di stabilire cosa tracciare, con che formato e con che livello di dettaglio.

L'accesso alla funzionalità di configurazione del servizio di auditing avviene tramite la voce *Auditing* nella sezione *Configurazione* del menu laterale sinistro.

Se la maschera si presenta come in Fig. 8.66 il servizio di auditing è disabilitato e quindi nessun dato verrà tracciato.

Modificando lo *Stato* del servizio di auditing in **Abilitato** appariranno ulteriori campi nel form (vedi Fig. 8.67) per effettuare le impostazioni.

La configurazione del servizio di auditing avviene tramite la creazione di una lista di **Filtri**, ciascuno dei quali stabilisce un criterio per stabilire se una data informazione deve o non deve essere tracciata. Alle informazioni cui non si applica nessuno dei filtri definiti, viene applicato il comportamento di default, i cui parametri sono presenti nella schermata principale del servizio. Facendo riferimento alla Fig. 8.67 vediamo quali sono i parametri per specificare il comportamento di default:

- **Audit** (abilitato/disabilitato): Se abilitato, tutte le informazioni, cui non risulta applicabile nessuno dei filtri impostati, verranno tracciate dal servizio di auditing.

Fig. 8.66: Servizio di auditing disabilitato

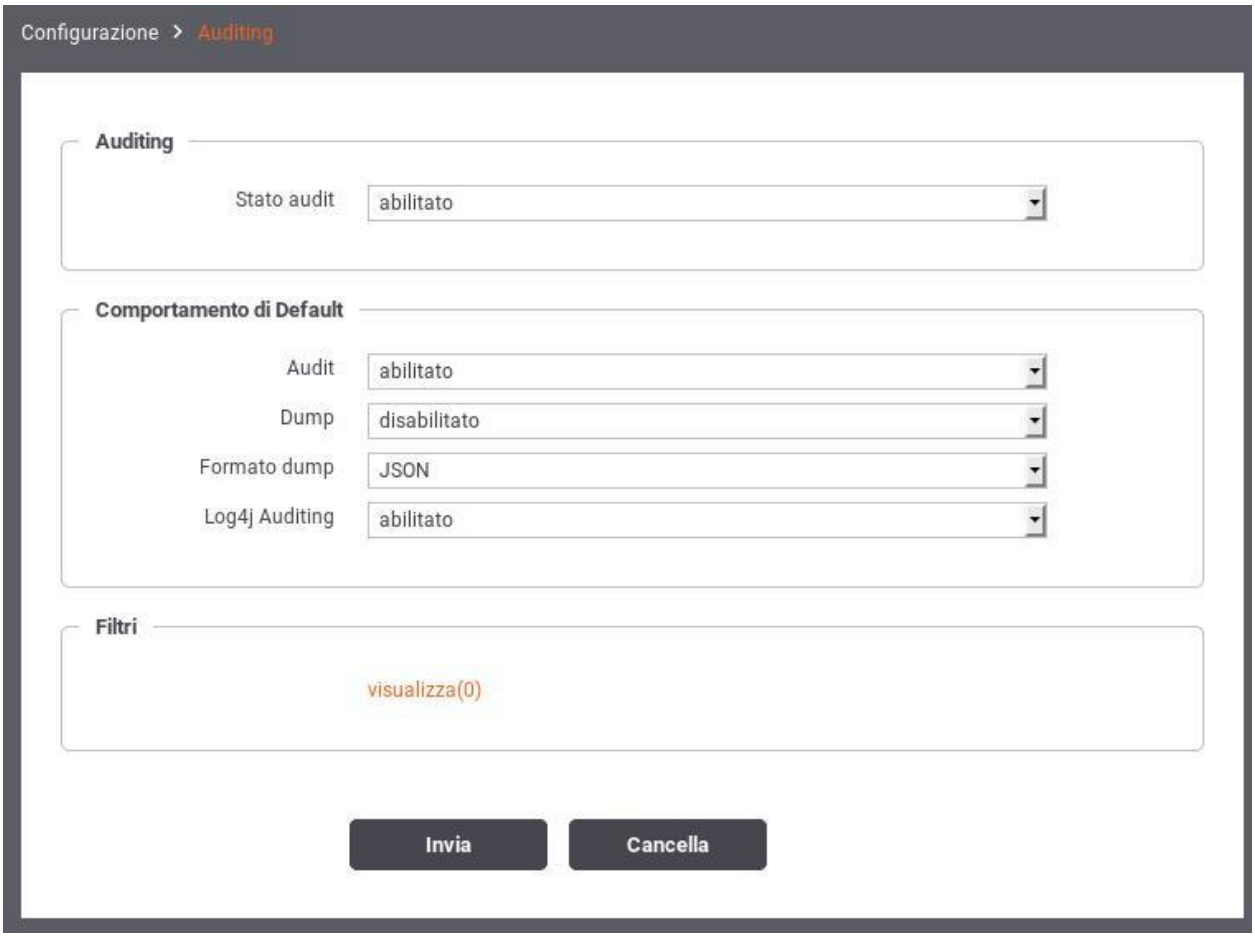


Fig. 8.67: Servizio di auditing abilitato

- **Dump** (abilitato/disabilitato): Questo campo viene preso in considerazione quando *Audit* = *abilitato*. Stabilisce, nei casi in cui non si applica nessun filtro, se oltre a tracciare i campi che descrivono l'operazione, devono essere tracciate anche le strutture dati coinvolte.
- **Formato Dump** (JSON/XML): Stabilisce il formato in cui vengono memorizzate le strutture dati di cui si è scelto di effettuare il dump. Le opzioni possibili sono tra il formato standard JSON (http://www.json.org) e la sua rappresentazione in formato XML.
- **Log4J Auditing** (abilitato/disabilitato): Questa opzione consente di abilitare/disabilitare l'appender log4j relativo ai dati tracciati dal servizio di auditing.

Una volta stabilito il comportamento di default si potranno definire i filtri specifici. Per passare alla sezione di gestione dei filtri si seleziona *Visualizza* nella sezione Filtri. Nell'area di gestione filtri viene mostrata la lista dei filtri esistenti con la possibilità di modificare/cancellare gli esistenti o inserirne di nuovi. Si può aggiungere un nuovo filtro premendo il pulsante *Aggiungi*. In Fig. 8.68 è mostrata la maschera per la creazione di un nuovo filtro di auditing.

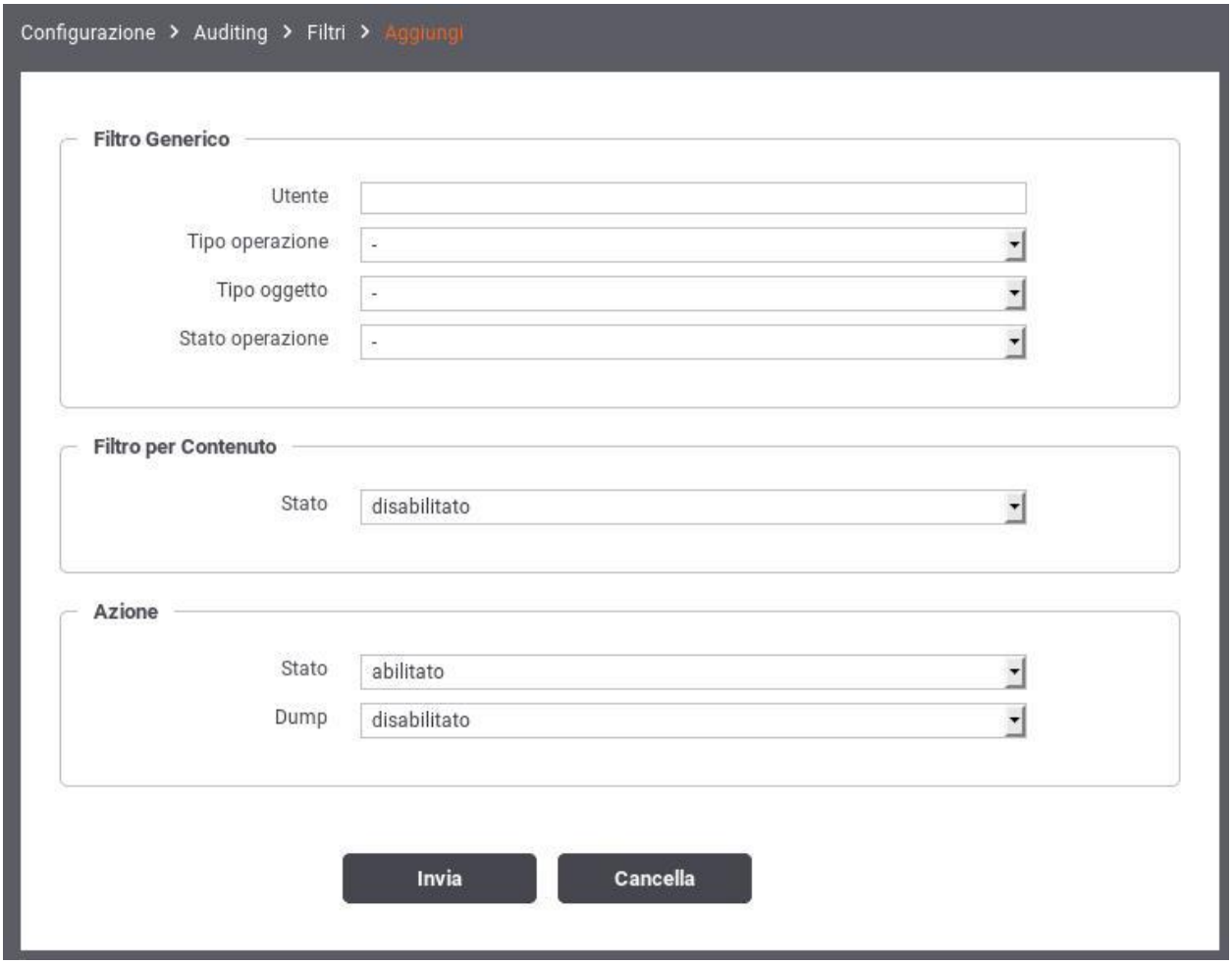


Fig. 8.68: Creazione di un filtro per il servizio di auditing

Facendo riferimento alla Fig. 8.68 vediamo in dettaglio il significato dei campi di un filtro:

- *Filtro Generico*

  - **Utente**: è possibile specificare in questo campo uno username relativo ad un utente della govwayConsole del quale si vogliono tracciare le operazioni effettuate. Lasciare il campo di testo vuoto equivale a *Qualsiasi Utente*
  - **Tipo Operazione** (ADD/CHANGE/DEL): Specifica il tipo di operazione che si vuole tracciare distinguendo tra operazioni di creazione, modifica e cancellazione. Lasciare il campo vuoto equivale a *Qualsiasi Tipo*.
  - **Tipo Oggetto**: Questo campo è costituito da una lista contenente tutte le entità gestibili tramite l'interfaccia govwayConsole (ad esempio: Accordo di Servizio, Porta Delegata, ecc). Consente di restringere il tracciamento alle sole operazioni riguardanti una determinata entità. Lasciare il campo vuoto equivale a *Qualsiasi Tipo Oggetto*.
  - **Stato Operazione** (requesting/error/completed): Consente di restringere le operazioni da tracciare in base al loro stato:
    * *requesting*: indica un'operazione in fase di richiesta e non ancora completata
    * *error*: Indica un'operazione completata che ha restituito un errore
    * *completed*: Indica un'operazione che è terminata correttamente

    Lasciare il campo vuoto equivale a *Qualsiasi Stato Operazione*.

- *Filtro per contenuto*
  - **Stato** (abilitato/disabilitato): Opzione che consente di abilitare il filtro basato sul contenuto degli oggetti coinvolti nell'operazione. Se l'opzione viene abilitata compariranno i 2 campi descritti ai passi successivi.
  - **Tipo** (normale/espressioneRegolare): Descrive se la stringa riportata nel campo Dump deve essere interpretata come pattern o come espressione regolare.
  - **Dump**: Campo di testo per inserire il pattern (o espressione regolare) sulla base del quale verranno filtrate le operazioni. Il sistema di auditing traccerà soltanto le operazioni che coinvolgeranno entità il cui contenuto corrisponde alla stringa specificata.
- *Azione*: indica quale azione deve essere effettuata al verificarsi delle condizioni del filtro
  - **Stato** (abilitato/disabilitato): Se abilitato, al verificarsi delle condizioni impostate nel filtro, i dati dell'operazione verranno tracciati.
  - **Dump** (abilitato/disabilitato): Se *Stato* = *abilitato* è possibile specificare se si deve effettuare anche il dump delle entità coinvolte nell'operazione. Ad esempio, se viene tracciata un'operazione di modifica di un Accordo di Servizio, si decide se si vuole effettuare anche il dump dell'Accordo di Servizio oggetto della modifica.

CAPITOLO 9

Errori generati da GovWay

La gestione dei casi di errore, nelle comunicazioni mediate da un Gateway, deve tener conto di ulteriori casi di errore che possono presentarsi rispetto al dialogo diretto tra gli applicativi. Oltre agli errori già previsti nelle interfacce dell'API, gli applicativi client possono pertanto ricevere due tipi di errori generati direttamente da GovWay:

- *Errori Client*: identificabili da un codice http 4xx su API REST o da un fault code "Client" su API SOAP. Indicano che GovWay ha rilevato problemi nella richiesta effettuata dal client (es. errore autenticazione, autorizzazione, validazione contenuti. . .).
- *Errori Server*: identificabili dai codici http 502, 503 e 504 per le API REST o da un fault code "Server" generato dal Gateway e restituito con codice http 500 per le API SOAP.

La codifica degli errori prodotta dal Gateway permette alle applicazioni client di discriminare tra errori causati da una richiesta errata, per i quali è quindi necessario intervenire sull'applicazione client prima di effettuare nuovi invii, ed errori dovuti allo stato dei servizi invocati, per i quali è invece possibile continuare ad effettuare la richiesta. Maggiori dettagli sulla logica di re-invio delle richieste vengono riportati nella sezione *Classificazione degli Errori*.

Per ciascun errore GovWay riporta le seguenti informazioni:

- Un codice http su API REST o un fault code su API SOAP come descritto in precedenza.
- Un codice di errore, indicato nell'header http "GovWay-Transaction-ErrorType", che riporta l'errore rilevato dal gateway (es. AuthenticationRequired, TokenExpired, InvalidRequestContent . . .).
- Un identificativo di transazione, indicato nell'header http "GovWay-Transaction-ID", che identifica la transazione in errore, utile principalmente per indagini diagnostiche.
- Un payload http, contenente maggiori dettagli sull'errore, opportunamente codificato per API REST (*REST Problem Details - RFC 7807*) o SOAP (*SOAP Fault*).

Nota: Il codice di errore e l'identificativo di transazione vengono riportati sia tramite header http che all'interno del payload.

Di seguito viene riportato un esempio di errore generato in seguito al rilevamento di una richiesta non conforme all'interfaccia API REST:

```
HTTP/1.1 400 Bad Request
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: application/problem+json
Date: Thu, 28 May 2020 15:59:14 GMT

{
    "type":"https://govway.org/handling-errors/400/InvalidRequestContent.html",
    "title":"InvalidRequestContent",
    "status":400,
    "detail":"Request content not conform to API specification",
    "govway_id":"b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd"
}
```

Lo stesso tipo di errore, rilevato per una API SOAP, viene riportato di seguito:

```
HTTP/1.1 500 Internal Server Error
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: text/xml
Date: Thu, 28 May 2020 15:59:14 GMT

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client.InvalidRequestContent</faultcode>
      <faultstring>Received request is not conform to API specification</faultstring>
      <faultactor>http://govway.org/integration</faultactor>
      <detail>
         <problem xmlns="urn:ietf:rfc:7807">
            <type>https://govway.org/handling-errors/400/InvalidRequestContent.html</
↪type>
            <title>InvalidRequestContent</title>
            <status>400</status>
            <detail>Request content not conform to API specification</detail>
            <govway_id>9876b03e-0377-4a02-9fb8-07094b0cdf06</govway_id>
         </problem>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

9.1 Classificazione degli Errori

Una risposta con codice 2xx indica che l'operazione ha avuto successo mentre codici diversi indicano un problema imputabile al client (4xx su API REST o da un fault code "Client" su API SOAP) o un errore dipendente dallo stato del servizio (5xx su API REST o da un fault code "Server" su API SOAP).

La tabella Tabella 9.1 riporta l'elenco dei possibili codici di errore restituiti da GovWay. Per ognuno di questi, nella colonna "Retry" è indicato se sia possibile o meno effettuare nuovi invii della stessa richiesta che ha ottenuto errore. Le indicazioni fornite sono le seguenti:

- Sì: il client può effettuare nuovamente la stessa richiesta;
- Sì, se idempotente: il client può effettuare nuovamente la stessa richiesta, ma solo se l'operazione sul backend applicativo è implementata in maniera idempotente.
- No: il client deve risolvere il problema segnalato prima di effettuare una nuova richiesta (ripetere la stessa richiesta produrrebbe sempre lo stesso esito).

Tabella 9.1: Gestione degli Errori

REST / SOAP	GovWay-Transaction-ErrorType	Retry
400 / Client	*Errori 400 (Bad Request)*	No
401 / Client	*Errori 401 (Authentication Error)*	No
403 / Client	*Errori 403 (Authorization Deny)*	No
404 / Client	*Errori 404 (NotFound)*	No
409 / Client	*Errori 409 (Conflict)*	No
413 / Client	*Errori 413 (Payload Too Large)*	No
429 / Client	*LimitExceeded* - 429 (Rate Limiting)	No, fino a reset *
429 / Client	*TooManyRequests* - 429 (Rate Limiting)	Sì
502 / Server	*Errori 502 (Bad Gateway)*	Sì, se idempotente
502 / Server	*ResponseSizeExceeded* - 502 (Bad Gateway)	No
503 / Server	*Errori 503 (Service Unavailable)*	Sì
504 / Server	*Errori 504 (Endpoint Request Timed-out)*	Sì, se idempotente

[*] Se vengono attivate policy di *Rate Limiting* che prevedono un limite di richieste all'interno di una finestra temporale, GovWay genera un header HTTP che indica al client il numero di secondi che mancano alla nuova finestra temporale dove saranno resettati i contatori delle richieste effettuate. I nomi degli header, che cambiano in funzione delle policy attivate, vengono descritte nella sezione *Informazioni restituite dal gateway nella risposta all'applicativo client*.

Nei casi in cui è prevista la rispedizione, GovWay genera un header "Retry-After" che indica al client il numero di secondi di attesa prima di ripetere la richiesta.

9.1.1 Errori 400 (Bad Request)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi ad una richiesta client malformata.

Nella configurazione di default di GovWay, le casistiche di errore "AttachmentsRequestFailed", "MessageSecurityRequestFailed", "InteroperabilityRequestManagementFailed", "TransformationRuleRequestFailed" e "ConnectorNotFound" sono tutte restituiti al client con il solo codice di errore *BadRequest*. La scelta è finalizzata ad evitare disclosure di informazioni relative al domino interno.

È possibile abilitare temporaneamente la generazione dei codici puntuali accendendo alla voce "Strumenti - Runtime" della console di gestione e selezionando "Errore Puntuale" per la "Richiesta" nella sezione «Errori generati dal Gateway - Codici di errore "GovWay-Transaction-ErrorType"» (Fig. 9.1).

L'abilitazione permanente può essere invece effettuata disabilitando la seguente proprietà sul file di proprietà esterno /etc/govway/errori_local.properties:

```
# Gateway non in grado di gestire la richiesta: AttachmentsRequestFailed,␣
↪MessageSecurityRequestFailed, InteroperabilityRequestManagementFailed,␣
↪TransformationRuleRequestFailed, ConnectorNotFound
WRAP_400_INTERNAL_BAD_REQUEST.enabled=false
```

Fig. 9.1: Attivazione temporanea degli errori specifici 400 (Bad Request)

ContentTypeNotProvided

GovWay ha rilevato una richiesta verso una API SOAP che non possiede un header http "Content-Type".

ContentTypeNotSupported

GovWay ha rilevato una richiesta verso una API SOAP che possiede un header http "Content-Type" non supportato.

Il valore supportato per SOAP 1.1 è "text/xml" mentre per SOAP 1.2 è "application/soap+xml". Sono supportati anche i formati multipart "SOAP With Attachments" (Multipart/Related; type=text/xml; boundary=...) e MTOM (Multipart/Related; type=»application/xop+xml»; start-info=»text/xml»; boundary=...).

SoapMustUnderstandUnknown

GovWay ha rilevato una richiesta verso una API SOAP che contiene un SOAP Header con attributo "mustUnderstand" e senza un actor/role definito che risulta sconosciuto a GovWay.

Nota: L'errore viene generato solamente se GovWay è stato configurato per riconoscere e trattare solamente alcuni SOAP Header specifici. La configurazione di default di govway è di far passare tutti i SOAP Header; per modificarla agire sul file di proprietà esterno /etc/govway/govway_local.properties aggiungendo le seguenti proprietà

```
# Possibili valori: true/false
org.openspcoop2.pdd.services.BypassMustUnderstandHandler.allHeaders=false

# Sintassi per filtri specifici:
# org.openspcoop2.pdd.services.BypassMustUnderstandHandler.header.LOCAL_
↪NAME=NAMESPACE_URI
# Se si deve definire più header con stesso local name e differente␣
↪namespace si può utilizzare la seguente sintassi:
# org.openspcoop2.pdd.services.BypassMustUnderstandHandler.header.LOCAL_NAME!
↪NUMERO_PROGRESSIVO=NAMESPACE_URI
# Esempio per Bypass per WS-Security:
#org.openspcoop2.pdd.services.BypassMustUnderstandHandler.header.
↪Security=http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
↪wssecurity-secext-1.0.xsd
# Esempio per Bypass per WS-Reliability
#org.openspcoop2.pdd.services.BypassMustUnderstandHandler.header.
↪Sequence=http://schemas.xmlsoap.org/ws/2005/02/rm
```

SoapVersionMismatch

La versione SOAP rilevata differisce tra quella indicata nell'header http "Content-Type" e il namespace dell'Envelope.

UnprocessableRequestContent

GovWay ha rilevato un payload differente da quello indicato nell'header http "Content-Type".

RequestReadTimeout

Rilevato errore "Read Timed Out" durante la lettura della richiesta.

NotSupportedByProtocol

L'errore viene sollevato quando la richiesta non è compatibile con il Profilo di Interoperabilità e/o il protocollo (SOAP/REST) a cui appartiene l'API invocata su GovWay.

CorrelationInformationNotFound

L'errore indica che nella richiesta non è stato possibile per GovWay estrarre il *Riferimento ID Richiesta* da utilizzare per effettuare una correlazione asincrona tra operazioni differenti.

La correlazione è attivabile su una API tramite la funzionalità descritta nella sezione *Correlazione tra transazioni differenti*.

Nota: In mancanza di un *Riferimento ID Richiesta*, GovWay per default non solleva alcun errore. È possibile forzare la generazione dell'errore intervenendo sul file di proprietà esterno /etc/govway/trasparente_local.properties aggiungendo le seguenti proprietà

```
# Fruizioni
org.openspcoop2.protocol.trasparente.pd.riferimentoIdRichiesta.required=true
# Erogazioni
org.openspcoop2.protocol.trasparente.pa.riferimentoIdRichiesta.required=true
```

L'errore viene anche sollevato se GovWay non rileva il riferimento alla richiesta nelle collaborazioni asincrone del Profilo di Interoperabilità SPCoop. Per maggiori dettagli si rimanda alla sezione *Profili Asincroni*.

ApplicationCorrelationIdentificationRequestFailed

La funzionalità di correlazione applicativa, abilitata sull'API invocata, non è riuscita ad estrarre l'informazione richiesta.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Correlazione Applicativa*.

InvalidRequestContent

La funzionalità di validazione dei contenuti applicativi, abilitata sull'API invocata, ha rilevato un contenuto della richiesta non conforme alla specifica dell'API.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Validazione dei messaggi*.

UnexpectedInteroperabilityHeader

GovWay ha rilevato, in una fruizione di API, una richiesta già contenente l'header di interoperabilità previsto dal profilo.

L'errore viene generato solo se l'API appartiene ad un Profilo di Interoperabilità differente dal profilo "API Gateway". Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

InteroperabilityInvalidRequest

L'errore segnala che GovWay ha rilevato una richiesta non conforme al Profilo di Interoperabilità a cui appartiene l'API invocata.

Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

AttachmentsRequestFailed

GovWay ha rilevato un payload della richiesta non utilizzabile con la gestione degli attachments abilitata sull'API.

Maggiori dettagli sulla funzionalità di gestione degli attachments sono presenti nella sezione *MTOM*.

MessageSecurityRequestFailed

GovWay ha rilevato un payload della richiesta non utilizzabile con la gestione della sicurezza messaggio abilitata sull'API.

Maggiori dettagli sulla funzionalità di sicurezza messaggio sono presenti nella sezione *Sicurezza a livello del messaggio*.

InteroperabilityRequestManagementFailed

L'errore segnala che GovWay non è riuscito a completare la generazione di un header conforme al Profilo di Interoperabilità a cui appartiene l'API invocata.

Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

TransformationRuleRequestFailed

GovWay ha rilevato un payload della richiesta non utilizzabile con la funzionalità di trasformazione attivata sull'API.

Maggiori dettagli sulla funzionalità di trasformazione sono presenti nella sezione *Trasformazioni*.

ConnectorNotFound

Non è stato possibile individuare il connettore che implementa l'API tra quelli associati all'erogazione.

Maggiori dettagli sulla funzionalità che consente di individuare il connettore da utilizzare, rispetto ai parametri della richiesta, sono presenti nella sezione *Consegna Condizionale*.

BadRequest

L'errore segnala che la richiesta verso l'API invocata è malformata.

9.1.2 Errori 401 (Authentication Error)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi a autenticazione fallita. Rientrano in questa casistica gli errori avenuti durante le fasi di autenticazione degli applicativi (Sezione *Autenticazione Trasporto*) e di verifica del token OAuth (Sezione *Autenticazione Token*).

AuthenticationRequired

La richiesta non possiede le credenziali relative all'autenticazione configurata sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autenticazione sono descritti nella sezione *Autenticazione Trasporto*.

AuthenticationFailed

La richiesta possiede delle credenziali non valide relative all'autenticazione configurata sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autenticazione sono descritti nella sezione *Autenticazione Trasporto*.

ProxyAuthenticationRequired

La richiesta non possiede le credenziali necessarie per poter autenticare il frontend su cui è stata effettuata l'autenticazione degli applicativi chiamanti.

Scenario in cui si presenta l'errore

Nel caso in cui la terminazione ssl viene gestita su un frontend http (Apache httpd, IIS, etc) GovWay necessita di ricevere le credenziali per attuare il processo di autenticazione descritto nella sezione *Autenticazione Trasporto*. Nel caso di utilizzo di una integrazione "mod_jk" tra frontend e application server, GovWay riceve i certificati gestiti sul frontend http in maniera trasparente. Negli altri casi invece deve essere configurato opportunamente il frontend http per inoltrare i certificati client o il DN attraverso header HTTP a GovWay.

Nell'ambito di tale configurazione è possibile abilitare l'autenticazione del frontend in modo da accettare gli header http contenenti le credenziali solamente da un frontend autenticato.

Maggiori dettagli sulla funzionalità sono descritti nella sezione install_ssl_server_frontend della Guida di Installazione.

ProxyAuthenticationFailed

La richiesta possiede delle credenziali non valide per poter autenticare il frontend su cui è stata effettuata l'autenticazione degli applicativi chiamanti.

Scenario in cui si presenta l'errore

Nel caso in cui la terminazione ssl viene gestita su un frontend http (Apache httpd, IIS, etc) GovWay necessita di ricevere le credenziali per attuare il processo di autenticazione descritto nella sezione *Autenticazione Trasporto*. Nel caso di utilizzo di una integrazione "mod_jk" tra frontend e application server, GovWay riceve i certificati gestiti sul frontend http in maniera trasparente. Negli altri casi invece deve essere configurato opportunamente il frontend http per inoltrare i certificati client o il DN attraverso header HTTP a GovWay.

Nell'ambito di tale configurazione è possibile abilitare l'autenticazione del frontend in modo da accettare gli header http contenenti le credenziali solamente da un frontend autenticato.

Maggiori dettagli sulla funzionalità sono descritti nella sezione install_ssl_server_frontend della Guida di Installazione.

ForwardProxyAuthenticationRequired

Il frontend che ha effettuato l'autenticazione degli applicativi non ha inoltrato a GovWay le credenziali nell'header http concordato.

Scenario in cui si presenta l'errore

Nel caso in cui la terminazione ssl viene gestita su un frontend http (Apache httpd, IIS, etc) GovWay necessita di ricevere le credenziali per attuare il processo di autenticazione descritto nella sezione *Autenticazione Trasporto*. Nel caso di utilizzo di una integrazione "mod_jk" tra frontend e application server, GovWay riceve i certificati gestiti sul frontend http in maniera trasparente. Negli altri casi invece deve essere configurato opportunamente il frontend http per inoltrare i certificati client o il DN attraverso header HTTP a GovWay.

Maggiori dettagli sulla funzionalità sono descritti nella sezione install_ssl_server_frontend della Guida di Installazione.

TokenAuthenticationRequired

La richiesta non possiede un token *oAuth2*. Il token viene richiesto dall'autenticazione configurata sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Autenticazione Token*.

TokenAuthenticationFailed

La richiesta possiede un token *oAuth2* non valido rispetto all'autenticazione configurata sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Autenticazione Token*.

TokenExpired

La richiesta possiede un token *oAuth2* scaduto.

Maggiori dettagli sulla funzionalità configurata nel Controllo degli Accessi dell'API sono descritti nella sezione *Autenticazione Token*.

TokenNotBefore

La richiesta possiede un token *oAuth2* non ancora utilizzabile. Nel claim "notBefore" è presenta una data futura.

Maggiori dettagli sulla funzionalità "Autenticazione Token" configurata nel Controllo degli Accessi dell'API sono descritti nella sezione *Autenticazione Token*.

TokenRequiredClaimsNotFound

Il token *oAuth2* presente nella richiesta non contiene tutti i claim configurati come obbligatori nel Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Autenticazione Token*.

Authentication

La richiesta non soddisfa l'autenticazione indicata nel Controllo degli Accessi dell'API (*Autenticazione Trasporto*).

9.1.3 Errori 403 (Authorization Deny)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay, relativi ad autorizzazione negata.

AuthorizationContentDeny

La richiesta non soddisfa i criteri di autorizzazione del contenuto attivati sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione dei contenuti sono descritti nella sezione *Autorizzazione Contenuti*.

AuthorizationContentPolicyDeny

La richiesta non soddisfa i criteri di autorizzazione del contenuto attivati, sul Controllo degli Accessi dell'API, tramite una policy XAML (o un template).

Maggiori dettagli sulla funzionalità di autorizzazione dei contenuti sono descritti nella sezione *Autorizzazione Contenuti*.

AuthorizationDeny

La richiesta non soddisfa i criteri di autorizzazione per richiedente, attivati sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione per richiedente sono descritti nella sezione *Autorizzazione*.

AuthorizationPolicyDeny

La richiesta non soddisfa i criteri di autorizzazione definiti nella policy XAML (o in un template), attivati sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione per policy sono descritti nella sezione *XACML-Policy*.

AuthorizationTokenDeny

La richiesta non soddisfa i criteri di autorizzazione, relativi ai claims presenti nel token *oAuth2*, attivati sul Controllo degli Accessi dell'API. Maggiori dettagli sulla funzionalità di autorizzazione per token claims sono descritti nella sezione *Autorizzazione per Token Claims*.

AuthorizationMissingScope

Il token *oAuth2* presente nella richiesta non contiene tutti gli scope richiesti sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione per scope sono descritti nella sezione *Autorizzazione*.

AuthorizationMissingRole

La richiesta non soddisfa i criteri di autorizzazione per ruolo, attivati sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione per ruolo sono descritti nella sezione *Autorizzazione*.

Authorization

La richiesta non soddisfa i criteri di autorizzazione attivati sul Controllo degli Accessi dell'API.

Maggiori dettagli sulla funzionalità di autorizzazione sono descritti nella sezione *Autorizzazione*.

9.1.4 Errori 404 (NotFound)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi a richieste verso API o risorse inesistenti.

UndefinedOperation

L'operazione richiesta non risulta associata all'API registrata su GovWay.

Maggiori dettagli sulla registrazione di una API su GovWay sono descritti nella sezione *Definizione delle API*.

UnknownAPI

L'API richiesta non risulta esistere su GovWay.

Maggiori dettagli sulla registrazione di una API su GovWay sono descritti nella sezione *Definizione delle API*.

NotFound

Il contenuto della richiesta non indirizza una API esistente su GovWay.

Questo tipo di errore avviene nei Profili di Interoperabilità per i quali l'API non viene indirizzata nella URL ma all'interno del contenuto della richiesta (es. *Profilo "SPCoop"*).

9.1.5 Errori 409 (Conflict)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi a richieste già processate.

ConflictInQueue

La richiesta risulta già in elaborazione su GovWay.

Questo tipo di errore avviene nei Profili di Interoperabilità che richiedono un filtro dei duplicati. Un identificativo univoco viene associato al messaggio all'interno dell'header previsto dal profilo di interoperabilità (es. *Profilo "ModI"*).

Conflict

La richiesta risulta già stata elaborata su GovWay.

Questo tipo di errore avviene nei Profili di Interoperabilità che richiedono un filtro dei duplicati. Un identificativo univoco viene associato al messaggio all'interno dell'header previsto dal profilo di interoperabilità (es. *Profilo "ModI"*).

9.1.6 Errori 413 (Payload Too Large)

In questa sezione vengono riportati i possibili codici di errore generati da GovWay relativi a richieste che non vengono processate poichè possiedono un payload più grande del livello di soglia impostato tramite le policy di *Rate Limiting* definite utilizzando la metrica "Dimensione Massima Messggio".

RequestSizeExceeded

La richiesta non viene processata poichè possiede un payload più grande del livello di soglia impostato tramite la policy di *Rate Limiting* definita utilizzando la metrica "Dimensione Massima Messaggio".

9.1.7 Errori 429 (Rate Limiting)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi alle funzionalità di Rate Limiting.

LimitExceeded

L'errore segnala che è stato superato il numero massimo di richieste (o di banda) nell'intervallo temporale configurato sulla policy di Rate Limiting dell'API invocata.

Maggiori informazioni sul Rate Limiting sono consultabili nella sezione *Rate Limiting*.

TooManyRequests

L'errore segnala che è stato superato il numero totale massimo di richieste simultanee permesse sull'API invocata.

Maggiori informazioni sul Rate Limiting sono consultabili nella sezione *Rate Limiting*.

9.1.8 Errori 502 (Bad Gateway)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi a errori emersi durante la gestione della risposta.

Nella configurazione di default di GovWay, gli errori descritti in questa sezione, con l'eccezione del codice «ResponseSizeExceeded», sono tutti restituiti al client con il solo codice di errore *InvalidResponse*. La scelta è finalizzata ad evitare disclosure di informazioni relative al domino interno.

È possibile abilitare temporaneamente la generazione dei codici puntuali accendendo alla voce "Strumenti - Runtime" della console di gestione e selezionando "Errore Puntuale" per la "Risposta" nella sezione «Errori generati dal Gateway - Codici di errore "GovWay-Transaction-ErrorType"» (Fig. 9.2).

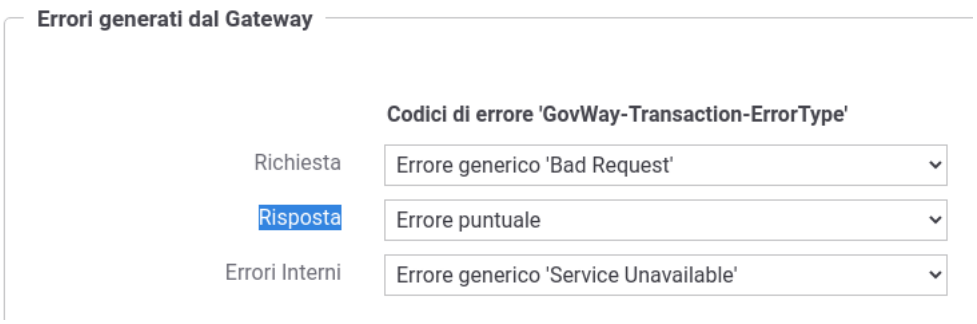


Fig. 9.2: Attivazione temporanea degli errori specifici 502 (Bad Gateway)

L'abilitazione permanente può essere invece effettuata disabilitando le seguenti proprietà sul file di proprietà esterno /etc/govway/errori_local.properties:

```
WRAP_502_BAD_RESPONSE.enabled=false
WRAP_502_INTERNAL_RESPONSE_ERROR.enabled=false
```

InvalidResponse

Risposta non valida ricevuta dal backend che implementa l'API.

ResponseSizeExceeded

La risposta non viene processata poichè possiede un payload più grande del livello di soglia impostato tramite la policy di *Rate Limiting* definita utilizzando la metrica "Dimensione Massima Messaggio".

UnprocessableResponseContent

GovWay ha rilevato un payload differente da quello indicato nell'header http "Content-Type" della risposta.

AttachmentsResponseFailed

GovWay ha rilevato un payload della risposta non utilizzabile con la gestione degli attachments abilitata sull'API.

Maggiori dettagli sulla funzionalità di gestione degli attachments sono presenti nella sezione *MTOM*.

ApplicationCorrelationIdentificationResponseFailed

La funzionalità di correlazione applicativa, abilitata sull'API invocata, non è riuscita ad estrarre l'informazione richiesta dalla risposta.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Correlazione Applicativa*.

MessageSecurityResponseFailed

GovWay ha rilevato un payload della risposta non utilizzabile con la gestione della sicurezza messaggio abilitata sull'API.

Maggiori dettagli sulla funzionalità di sicurezza messaggio sono presenti nella sezione *Sicurezza a livello del messaggio*.

InvalidResponseContent

La funzionalità di validazione dei contenuti applicativi, abilitata sull'API invocata, ha rilevato un contenuto della risposta non conforme alla specifica dell'API.

Maggiori dettagli sulla funzionalità sono descritti nella sezione *Validazione dei messaggi*.

InteroperabilityResponseManagementFailed

L'errore segnala che GovWay non è riuscito a completare la generazione di un header conforme al Profilo di Interoperabilità a cui appartiene l'API invocata.

Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

InteroperabilityInvalidResponse

L'errore segnala che GovWay ha rilevato una risposta non conforme al Profilo di Interoperabilità a cui appartiene l'API invocata.

Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

UnexpectedInteroperabilityResponseHeader

GovWay ha rilevato, in una erogazione di API, una risposta già contenente l'header di interoperabilità previsto dal profilo.

L'errore viene generato solo se l'API appartiene ad un Profilo di Interoperabilità differente dal profilo "API Gateway". Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

InteroperabilityResponseError

L'errore segnala la ricezione di una risposta, conforme al Profilo di Interoperabilità, che segnala degli errori rilevati dalla controparte.

Per maggiori dettagli sui profili di interoperabilità è possibile consultare le sezioni:

- *Profilo "ModI"*
- *Profilo "eDelivery"*
- *Profilo "SPCoop"*
- *Profilo "eDelivery"*

TransformationRuleResponseFailed

GovWay ha rilevato un payload della risposta non utilizzabile con la funzionalità di trasformazione attivata sull'API.

Maggiori dettagli sulla funzionalità di trasformazione sono presenti nella sezione *Trasformazioni*.

ExpectedResponseNotReceived

Una risposta non è presente nel payload ritornato dal backend che implementa l'API.

ConflictResponse

La risposta risulta già stata elaborata su GovWay.

Questo tipo di errore avviene nei Profili di Interoperabilità che richiedono un filtro dei duplicati. Un identificativo univoco viene associato al messaggio all'interno dell'header previsto dal profilo di interoperabilità (es. *Profilo "ModI"*).

BadResponse

Risposta non valida ricevuta dal backend che implementa l'API.

GatewayError

Il gateway non è momentaneamente in grado di gestire la risposta.

9.1.9 Errori 503 (Service Unavailable)

In questa sezione vengono riportati tutti i possibili codici di errore generati da GovWay relativi ad errori emersi durante la gestione della richiesta. Gli errori sono classificabili in:

- indisponibilità temporanea del backend che implementa l'API; l'errore viene identificato dal codice di errore "APIUnavailable";
- accesso all'API sospeso su GovWay; l'errore viene identificato dal codice di errore "APISuspended";
- Il gateway non è al momento correttamente operativo; rientrano in questa casistica i codici di errore: "GatewayInactive", "GatewayUnavailable", "GatewayError".

Nella configurazione di default di GovWay, gli errori che indicano una indisponibilità temporanea del gateway sono tutti restituiti al client con il solo codice di errore *APIUnavailable*. La scelta è finalizzata ad evitare disclosure di informazioni relative al domino interno.

È possibile abilitare temporaneamente la generazione dei codici puntuali accendendo alla voce "Strumenti - Runtime" della console di gestione e selezionando "Errore Puntuale" per gli "Errori Interni" nella sezione «Errori generati dal Gateway - Codici di errore "GovWay-Transaction-ErrorType"» (Fig. 9.3).

Fig. 9.3: Attivazione temporanea degli errori specifici 503 (Service Unavailable)

L'abilitazione permanente può essere invece effettuata disabilitando la seguente proprietà sul file di proprietà esterno /etc/govway/errori_local.properties:

```
# Gateway momentaneamente indisponibile: GatewayInactive, GatewayUnavailable,
↪ GatewayError
WRAP_503_INTERNAL_ERROR.enabled=false
```

APIUnavailable

GovWay ha rilevato una indisponibilità temporanea del backend che implementa l'API.

È possibile utilizzare la funzionalità descritta nella sezione *Verifica Connettività Connettore* per verificare puntualmente la connettività tramite la console di gestione.

APISuspended

L'API invocata risulta sospesa.

Maggiori dettagli sulla funzionalità di sospensione di una API vengono forniti nella sezione *Sospensione API*.

GatewayInactive

Il gateway non è attualmente disponibile.

GatewayUnavailable

Il gateway è temporaneamente non disponibile.

GatewayError

Il gateway non è momentaneamente in grado di gestire la richiesta.

9.1.10 Errori 504 (Endpoint Request Timed-out)

In questa sezione è presente il codice di errore generato da GovWay quando avviene un "Read Timed Out" durante l'invocazione del backend che implementa l'API.

EndpointReadTimeout

Rilevato errore "Read Timed Out" durante l'invocazione del backend che implementa l'API.

È possibile configurare il tempo di attesa di una risposta agendo sui parametri di configurazione del connettore descritti nella sezione *Tempi Risposta*.

9.2 REST Problem Details - RFC 7807

Negli errori generati da GovWay, relativi alla gestione di richieste verso API di tipo REST, il payload http ritornato al chiamante contiene un oggetto *Problem Details* come definito nella specifica *RFC 7807* (https://tools.ietf.org/html/rfc7807).

Gli elementi valorizzati sono i seguenti:

- *type*: riferisce una pagina della seguente documentazione (*Classificazione degli Errori*) che descrive l'errore.
- *title*: contiene un codice che specifica la problematica rilevata dal gateway (es. AuthenticationRequired, TokenExpired, InvalidRequestContent . . .). Tutti i codici di errore vengono descritti nella sezione *Classificazione degli Errori*.
- *status*: contiene il codice http ritornato al chiamante.
- *detail*: fornisce informazioni di dettaglio sull'errore avvenuto.
- *govway_id*: identificativo di transazione che permette di individuare la transazione terminata in errore tramite la Console di Monitoraggio.

Di seguito viene riportato un esempio:

```
{
    "type":"https://govway.org/handling-errors/400/InvalidRequestContent.html",
    "title":"InvalidRequestContent",
    "status":400,
    "detail":"Request content not conform to API specification",
    "govway_id":"b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd"
}
```

L'oggetto *Problem Details* generato dal Gateway possiede per default il formato *json*.

Viene utilizzato il formato *xml* (Appendice "A" del RFC 7807) solamente se la richiesta presenta anch'essa tale formato.

Nota: Un applicativo client può indicare al Gateway quale formato desidera attraverso l'header http *Accept*.

Di seguito viene riportato un esempio di oggetto *Problem Details* nel formato xml:

```
<problem xmlns="urn:ietf:rfc:7807">
    <type>https://govway.org/handling-errors/400/InvalidRequestContent.html</type>
    <title>InvalidRequestContent</title>
    <status>400</status>
    <detail>Request content not conform to API specification</detail>
    <govway_id>a1e047bf-3775-4f74-9492-dba972e7afb2</govway_id>
</problem>
```

Claim aggiuntivi

Se viene abilitata la generazione di un codice specifico di errore, come descritto nella sezione *Attivazione di Codici di Errore Specifici*, viene valorizzato anche il claim *govway_status*.

È inoltre possibile valorizzare il claim *instance* con l'identificativo dell'erogazione o della fruizione invocata seguendo le indicazioni descritte di seguito. Per default tale elemento non viene valorizzato.

È possibile abilitare temporaneamente la valorizzazione del claim *instance* accendendo alla voce "Strumenti - Runtime" della console di gestione e abilitando lo stato della sezione «Claim "instance" nei Problem " (Fig. 9.5)».

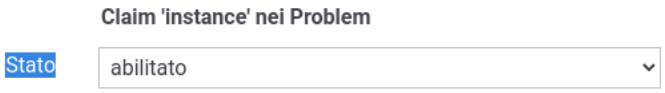


Fig. 9.4: Attivazione temporanea del claim *instance* nel Problem Detail

Una abilitazione permanente è invece attuabile agendo sul file di proprietà esterno /etc/govway/govway_local.properties abilitando la seguente proprietà:

```
org.openspcoop2.pdd.errori.instance=true
```

Di seguito viene riportato un esempio di errore generato in seguito al rilevamento di una richiesta non conforme all'interfaccia API REST, dove è stato abilitata sia la generazione di un codice di errore specifico che la valorizzazione dell'elemento instance:

```
{
    "type":"https://govway.org/handling-errors/400/InvalidRequestContent.html",
    "title":"InvalidRequestContent",
    "status":400,
    "detail":"Request content not conform to API specification",
    "instance":"gw_ENTE/gw_api-monitor/v1",
    "govway_id":"b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd",
    "govway_status":"integration:GOVWAY-418"
}
```

9.3 SOAP Fault

Negli errori generati da GovWay, relativi alla gestione di richieste verso API di tipo SOAP, il payload http ritornato al chiamante contiene un SOAP Fault.

Gli elementi del fault sono valorizzati come segue:

- *faultactor* (Soap 1.1) o *Role* (Soap 1.2) possiede il valore *http://govway.org/integration*.
- *faultcode* (Soap 1.1) o *Code/Subcode* (Soap 1.2): contiene uno standard SOAP fault code (Server/Client per Soap 1.1, Receiver/Sender per Soap 1.2) concatenato con un codice di errore di GovWay che specifica la problematica rilevata (es. AuthenticationRequired, TokenExpired, InvalidRequestContent ...). Tutti i codici di errore vengono descritti nella sezione *Classificazione degli Errori*.
- *faultstring* (Soap 1.1) o *Reason* (Soap 1.2): fornisce informazioni di dettaglio sull'errore avvenuto.
- *detail*: è presente l'oggetto *Problem Details*, nella rappresentazione xml descritta nella sezione *REST Problem Details - RFC 7807*.

Nota: Il formato di errore (*Soap 1.1* o *Soap 1.2*) assume lo stesso formato della richiesta.

Di seguito viene riportato un esempio di errore rilevato per una API SOAP 1.1:

```
HTTP/1.1 500 Internal Server Error
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: text/xml
Date: Thu, 28 May 2020 15:59:14 GMT

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client.InvalidRequestContent</faultcode>
      <faultstring>Received request is not conform to API specification</faultstring>
      <faultactor>http://govway.org/integration</faultactor>
      <detail>
         <problem xmlns="urn:ietf:rfc:7807">
            <type>https://govway.org/handling-errors/400/InvalidRequestContent.html</
↪type>
            <title>InvalidRequestContent</title>
            <status>400</status>
            <detail>Request content not conform to API specification</detail>
            <govway_id>9876b03e-0377-4a02-9fb8-07094b0cdf06</govway_id>
         </problem>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Lo stesso tipo di errore, rilevato per una API SOAP 1.2, viene riportato di seguito:

```
HTTP/1.1 500 Internal Server Error
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
```

(continues on next page)

(continua dalla pagina precedente)

```
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: application/soap+xml
Date: Thu, 28 May 2020 15:59:14 GMT

<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Body>
    <env:Fault>
       <env:Code>
          <env:Value>env:Sender</env:Value>
          <env:Subcode>
            <env:Value xmlns:integration="http://govway.org/integration/fault">
               integration:InvalidRequestContent
            </env:Value>
          </env:Subcode>
       </env:Code>
       <env:Reason>
          <env:Text xml:lang="en-US">Operation undefined in the API specification</
↪env:Text>
       </env:Reason>
       <env:Role>http://govway.org/integration</env:Role>
       <env:Detail>
         <problem xmlns="urn:ietf:rfc:7807">
            <type>https://govway.org/handling-errors/400/InvalidRequestContent.html</
↪type>
            <title>InvalidRequestContent</title>
            <status>400</status>
            <detail>Request content not conform to API specification</detail>
            <govway_id>9876b03e-0377-4a02-9fb8-07094b0cdf06</govway_id>
         </problem>
      </env:Detail>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

9.4 Attivazione di Codici di Errore Specifici

Nella configurazione di default di GovWay, gli errori restituiti ai client non contengono dettagli che possano causare disclosure di informazioni relative al dominio interno. In alcuni casi, per facilitare il supporto alla risoluzione di problemi, è comunque possibile abilitare la generazione di codici più specifici di errore ritornati al client nell'header http "GovWay-Transaction-ErrorStatus" e nel claim "govway_status" del *REST Problem Details - RFC 7807*.

È possibile abilitare temporaneamente la generazione dei codici specifici accendendo alla voce "Strumenti - Runtime" della console di gestione e abilitando "Http Header / Problem Detail" nella sezione «Codici di errore specifici "GovWay-Transaction-ErrorStatus" (Fig. 9.5)».

L'abilitazione permanente è invece possibile abilitando la seguente proprietà sul file di proprietà esterno /etc/govway/govway_local.properties:

```
org.openspcoop2.pdd.errori.status=true
```

Di seguito viene riportato un esempio di errore generato in seguito al rilevamento di una richiesta non conforme all'interfaccia API REST, dove è stata abilitata la generazione di un codice di errore specifico:

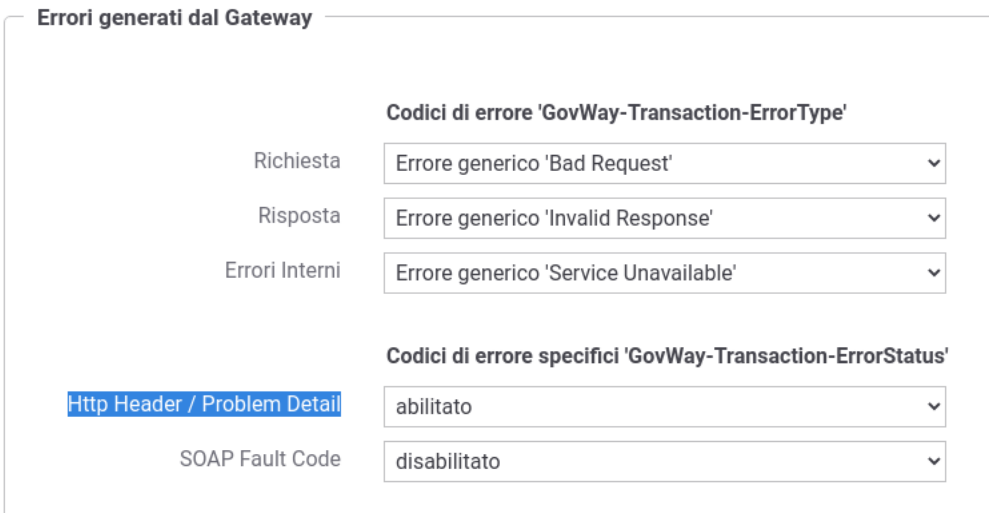


Fig. 9.5: Attivazione temporanea dei codici di errore specifici di GovWay

```
HTTP/1.1 400 Bad Request
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
GovWay-Transaction-ErrorStatus: integration:GOVWAY-418
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: application/problem+json
Date: Thu, 28 May 2020 15:59:14 GMT

{
    "type":"https://govway.org/handling-errors/400/InvalidRequestContent.html",
    "title":"InvalidRequestContent",
    "status":400,
    "detail":"Request content not conform to API specification",
    "govway_id":"b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd",
    "govway_status":"integration:GOVWAY-418"
}
```

Il codice di errore specifico può essere generato anche all'interno del SOAP Fault come "Fault Code" al posto di quello di default generato da GovWay e descritto nella sezione *Classificazione degli Errori*.

È possibile abilitare temporaneamente la generazione all'interno del SOAP Fault Code accendendo alla voce "Strumenti - Runtime" della console di gestione e abilitando "SOAP Fault Code" nella sezione «Codici di errore specifici "GovWay-Transaction-ErrorStatus"» (Fig. 9.6).

L'abilitazione permanente è invece possibile abilitando la seguente proprietà sul file di proprietà esterno /etc/govway/govway_local.properties:

```
org.openspcoop2.pdd.errori.soap.useGovWayStatusAsFaultCode=true
```

Di seguito viene riportato un esempio di errore generato in seguito al rilevamento di una richiesta non conforme all'interfaccia API SOAP, dove è stata abilitata sia la generazione di un codice di errore specifico che la generazione del SOAP Fault Code specifico:

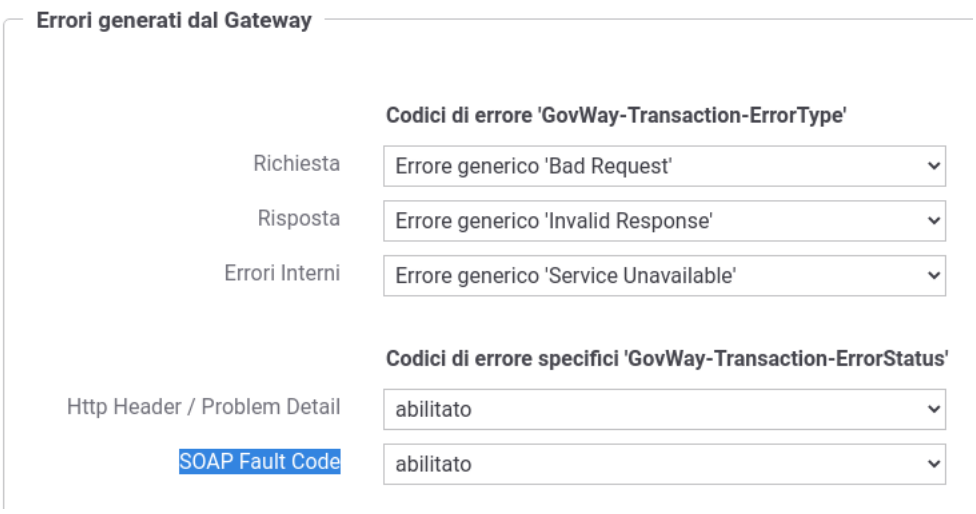


Fig. 9.6: Attivazione temporanea dei codici di errore specifici di GovWay come SOAP Fault Code

```
HTTP/1.1 500 Internal Server Error
Server: GovWay
Transfer-Encoding: chunked
GovWay-Transaction-ErrorType: InvalidRequestContent
GovWay-Transaction-ErrorStatus: integration:GOVWAY-418
GovWay-Transaction-ID: b76b4d1b-cd9d-43a0-bea2-1f352f1e71dd
Content-Type: text/xml
Date: Thu, 28 May 2020 15:59:14 GMT

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode xmlns:integration="http://govway.org/integration/fault">
            integration:Client.GOVWAY-423
      </faultcode>
      <faultstring>Received request is not conform to API specification</faultstring>
      <faultactor>http://govway.org/integration</faultactor>
      <detail>
         <problem xmlns="urn:ietf:rfc:7807">
            <type>https://govway.org/handling-errors/400/InvalidRequestContent.html</
↪type>
            <title>InvalidRequestContent</title>
            <status>400</status>
            <detail>Request content not conform to API specification</detail>
            <govway_id>9876b03e-0377-4a02-9fb8-07094b0cdf06</govway_id>
            <govway_status>integration:GOVWAY-418</govway_status>
         </problem>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Di seguito vengono riportate le casistiche di errore che possono verificarsi sul Gateway, con i relativi codici.

Nota: Alcuni degli errori riportati sono scaturiti da funzionalità disponibili nel Gateway attraverso configurazioni

avanzate non descritte nel presente manuale.

Tabella 9.2: Codici di Errore GovWay

Codice	Descrizione
integration:GOVWAY-401	Identifica la richiesta di una erogazione o fruizione inesistente
integration:GOVWAY-402	Rientrano in questa casistica gli errori avenuti durante la fase di autenticazione di una fruizione (sezione *Autenticazione Trasporto*)
integration:GOVWAY-403	Azione non identificabile tramite i meccanismi configurati. (sezione *Modalità di identificazione dell'azione*)
integration:GOVWAY-404	Rientrano in questa casistica gli errori avenuti durante la fase di autorizzazione di una fruizione (sezione *Autorizzazione*)
integration:GOVWAY-405	Servizio richiesto non esistente (richiede una configurazione non documentata)
integration:GOVWAY-406	Indica che non sono disponibili messaggi (richiede accesso alla MessageBox via Integration Manager, configurazione non documentata)
integration:GOVWAY-407	Il messaggio richiesto non esiste (richiede accesso alla MessageBox via Integration Manager, configurazione non documentata)
integration:GOVWAY-408	Indica che non esiste una API utilizzabile per correlare la richiesta con una precedente transazione (es. utilizzato con i profili asincroni descritti nella sezione *Profili Asincroni*)
integration:GOVWAY-409	Indica che non è possibile correlare la richiesta con una precedente transazione (es. utilizzato con i profili asincroni descritti nella sezione *Profili Asincroni*)
integration:GOVWAY-410	L'API invocata possiede il profilo *asincrono simmetrico* e la configurazione della fruizione non presenta meccanismi di autenticazione dell'applicativo client. L'identificazione di un applicativo fruitore è fondamentale nel profilo asincrono simmetrico per consegnare la risposta (*Profili Asincroni*)
integration:GOVWAY-411	Indica una configurazione errata dove l'applicativo mittente non possiede una configurazione per la spedizione della risposta asincrona e l'API possiede il profilo *asincrono simmetrico* (*Profili Asincroni*)
integration:GOVWAY-412	L'API è stata invocata senza fornire il riferimento ad un messaggio attualmente in carico sul Gateway, mentre la configurazione lo richiede. richiede una configurazione non documentata)
integration:GOVWAY-413	L'API è stata invocata fornendo il riferimento ad un messaggio attualmente in carico sul Gateway, mentre la configurazione non lo richiede. richiede una configurazione non documentata)
integration:GOVWAY-414	L'API invocata è stata configurata con un profilo differente da *oneway* e richiede la funzionalità di *consegna in ordine* (sezione *Profili di gestione della busta eGov*)
integration:GOVWAY-415	L'API invocata è stata configurata per utilizzare la funzionalità di *consegna in ordine* ma non presenta altre caratteristiche obbligatorie con questa funzionalità (es. confermaRicezione,filtroDuplicati,collaborazione) (sezione *Profili di gestione della busta eGov*)
integration:GOVWAY-416	Rientrano in questa casistica gli errori avenuti durante la fase di correlazione applicativa della richiesta (sezione *Correlazione Applicativa*)
integration:GOVWAY-417	Tale errore viene sollevato se l'interfaccia API e/o gli schemi associati (xsd,json,yaml) contengono errori che non ne consentono l'utilizzo durante la validazione dei contenuti (sezione *Validazione dei messaggi*)
integration:GOVWAY-418	La validazione dei contenuti ha rilevato una richiesta non conforme all'interfaccia API (sezione *Validazione dei messaggi*)

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
integration:GOVWAY-419	La validazione dei contenuti ha rilevato una risposta non conforme all'interfaccia API (sezione *Validazione dei messaggi*)
integration:GOVWAY-420	Viene sollevato questo errore se un applicativo invoca una fruizione di una API fornendo un messaggio contenente già un header di protocollo. (es. se viene inviato un messaggio contenente un'header eGov (sezione *Profilo "SPCoop"*)
integration:GOVWAY-421	Indica che il messaggio di richiesta fornito via Integration Manager non è un messaggio SOAP Valido (configurazione non documentata)
integration:GOVWAY-422	Il messaggio di richiesta presente nell'http body (Accesso al servizio out/xml2soap) o il messaggio indicato nella richiesta via Integration-Manager (Accesso al servizio via Integration Manager con imbustamento SOAP) non è utilizzabile, tramite la funzionalità di Imbustamento, per ottenere un messaggio SOAP valido (configurazione non documentata)
integration:GOVWAY-423	L'azione identificata tramite i meccanismi configurati non risulta esistere all'interno dell'API invocata. (sezione *Modalità di identificazione dell'azione*)
integration:GOVWAY-424	La funzionalità avanzata *Allega Body* ha generato un errore (configurazione non documentata)
integration:GOVWAY-425	La funzionalità avanzata *Scarta Body* ha generato un errore (configurazione non documentata)
integration:GOVWAY-426	Errore generico che può avvenire durante la gestione della richiesta, dovuto comunque a dati forniti nella richiesta stessa (es. Valore SOAPAction scorretto)
integration:GOVWAY-427	Indica che il Gateway ha rilevato la presenza di SOAPHeader Element che non è in grado di processare e che richiedono obbligatoriamente il processamento (mustUnderstand=1 e actor non presente)
integration:GOVWAY-428	Rientrano in questa casistica gli errori avenuti durante la fase di autorizzazione del contenuto (configurazione non documentata)
integration:GOVWAY-429	Errore che viene ritornato dal Gateway se la richiesta presenta un header http *Content-Type* non supportato (per API SOAP)
integration:GOVWAY-430	Errore che viene ritornato dal Gateway se rileva una busta soap che possiede un namespace differente da quello atteso per la versione SOAP corrispondente al *Content-Type* (per API SOAP)
integration:GOVWAY-431	Rientrano in questa casistica gli errori avvenuti durante il recupero delle credenziali fornite tramite un Proxy (configurazione non documentata)
integration:GOVWAY-432	Errore che viene ritornato dal Gateway se la richiesta presenta un contenuto malformato (es. xml malformato in una API SOAP)
integration:GOVWAY-433	Indica che la richiesta non presenta un header http *Content-Type* (obbligatorio in API SOAP)
integration:GOVWAY-434	Rientrano in questa casistica gli errori avenuti durante la fase di correlazione applicativa della risposta (sezione *Correlazione Applicativa*)
integration:GOVWAY-435	L'errore viene sollevato se viene rilevata una configurazione *Local Forward* non corretta (configurazione non documentata)
integration:GOVWAY-436	L'errore viene sollevato se viene rilevato un tipo di fruitore non supportato dalla modalità di utilizzo del Gateway fruita (configurazione non documentata)
integration:GOVWAY-437	L'errore viene sollevato se viene rilevato un tipo di erogatore non supportato dalla modalità di utilizzo del Gateway fruita (configurazione non documentata)

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
integration:GOVWAY-438	L'errore viene sollevato se viene rilevato un tipo di servizio non supportato dalla modalità di utilizzo del Gateway fruita (configurazione non documentata)
integration:GOVWAY-439	L'errore viene sollevato se viene rilevata una configurazione che richiede una funzionalità non supportata nella modalità di utilizzo del Gateway (configurazione non documentata)
integration:GOVWAY-440	Errore che viene ritornato dal Gateway se la risposta presenta un contenuto malformato (es. xml malformato in una API SOAP)
integration:GOVWAY-441	La richiesta indirizza una configurazione non invocabile direttamente, configurazione creata tramite le indicazioni descritte nella sezione *Differenziare le configurazioni specifiche per risorsa/azione*
integration:GOVWAY-442	La richiesta pervenuta sul Gateway non presenta un riferimento ad una precedente transazione, mentre la configurazione lo richiede (sezione *Correlazione tra transazioni differenti*). Nell'installazione di default del Gateway, l'errore indicato non viene mai sollevato poichè non è obbligatorio fornire il riferimento ad una precedente transazione.
integration:GOVWAY-443	L'errore viene ritornato dal Gateway se non viene rilevato un token durante l'invocazione di una fruizione (sezione *Autenticazione Token*)
integration:GOVWAY-444	L'errore viene ritornato dal Gateway se viene rilevato un token non valido durante l'invocazione di una fruizione (sezione *Autenticazione Token*)
integration:GOVWAY-445	Rientrano in questa casistica gli errori avenuti durante la fase di autorizzazione di una fruizione, riguardanti la gestione di un token (sezione *Autorizzazione*)
integration:GOVWAY-446	Il Gateway ritorna tale codice se la fruizione o l'erogazione invocata risulta sospesa
integration:GOVWAY-450	La richiesta pervenuta sul Gateway non indirizza una erogazione specifica e non è utilizzabile per identificarne alcuna (configurazione non documentata)
integration:GOVWAY-451	Il soggetto invocato non esiste (configurazione non documentata)
integration:GOVWAY-452	Indica che il messaggio ricevuto è già stato gestito in precedenza (es. filtro duplicati attivo descritto nella sezione *Profilo "SPCoop"*)
integration:GOVWAY-453	L'applicativo erogatore associato all'erogazione non esiste (configurazione non documentata)
integration:GOVWAY-454	Viene sollevato questo errore se il messaggio ritornato come risposta dall'applicativo erogatore, in una erogazione, contiene già un header di protocollo. (es. se viene ritornato un messaggio contenente un'header eGov (sezione *Profilo "SPCoop"*)
integration:GOVWAY-455	L'errore indica che la richiesta presenta al suo interno degli identificativi di API differenti da quelli dell'erogazione invocata (es. busta eGov contiene dei dati di servizio non allineati all'erogazione invocata)
integration:GOVWAY-500	Errore generico
integration:GOVWAY-516	Errore ritornato dal gateway se non riesce ad inoltrare il messaggio all'endpoint configurato
integration:GOVWAY-517	Errore ritornato dal gateway se non viene ritornata una risposta dall'endpoint contattato e il profilo ne prevede una (es. profilo sincrono nelle API SOAP)
integration:GOVWAY-518	Indica che l'applicativo erogatore ha ritornato un SOAPFault (API SOAP)
integration:GOVWAY-537	La richiesta pervenuta è già presente in carico sul Gateway ed è attualmente in fase di processamento (configurazione non documentata)

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
integration:GOVWAY-538	La richiesta asincrona pervenuta è già presente in carico sul Gateway ed è attualmente in fase di processamento (configurazione non documentata)
integration:GOVWAY-539	La ricevuta della richiesta asincrona pervenuta è già presente in carico sul Gateway ed è attualmente in fase di processamento (configurazione non documentata)
integration:GOVWAY-CC00	Errore generico avvenuto durante la gestione del Controllo del Traffico (sezione *Controllo del Traffico*)
integration:GOVWAY-CC01	Il Gateway ha rilevato il superamento del massimo numero di richieste simultanee configurato (sezione *Limitazione Numero di Richieste Complessive*)
integration:GOVWAY-CP00	Indica che la funzionalità di Rate-Limiting ha rilevato una policy sconosciuta (sezione *Rate Limiting*)
integration:GOVWAY-CP01	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "NumeroRichieste-RichiesteSimultanee" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP01	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "NumeroRichieste-RichiesteSimultanee" (sezione *Rate Limiting*).
integration:GOVWAY-CP02	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "NumeroRichieste-ControlloRealtime*" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado (sezione *Rate Limiting*).
integration:GOVWAY-ERR-CP02	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "NumeroRichieste-ControlloRealtime*" (sezione *Rate Limiting*).
integration:GOVWAY-CP03	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "OccupazioneBanda-*" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado (sezione *Rate Limiting*).
integration:GOVWAY-ERR-CP03	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "OccupazioneBanda-*" (sezione *Rate Limiting*).
integration:GOVWAY-CP04	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "TempoComplessivioRisposta" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP04	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "TempoComplessivioRisposta" (sezione *Rate Limiting*).
integration:GOVWAY-CP05	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "TempoMedioRisposta-*" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP05	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "TempoMedioRisposta-*" (sezione *Rate Limiting*).

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
integration:GOVWAY-CP06	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "NumeroRichiesteCompletateConSuccesso" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP06	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "NumeroRichiesteCompletateConSuccesso" (sezione *Rate Limiting*).
integration:GOVWAY-CP07	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "NumeroRichiesteFallite" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP07	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "NumeroRichiesteFallite" (sezione *Rate Limiting*).
integration:GOVWAY-CP08	Indica che la funzionalità di Rate-Limiting ha rilevato una violazione di una policy di tipo "NumeroFaultApplicativi" (sezione *Rate Limiting*). Il codice di errore può presentare il suffisso *-CC* se la policy è configurata insieme a controlli di congestione e/o il suffisso *-DP* se configurata con meccanismi di degrado.
integration:GOVWAY-ERR-CP08	Errore emerso durante la gestione da parte del Gateway della policy di Rate-Limiting di tipo "NumeroFaultApplicativi" (sezione *Rate Limiting*).
protocol:GOVWAY-109	Rientrano in questa casistica gli errori avenuti durante la fase di autenticazione di una erogazione, se non vengono rilevate credenziali (sezione *Autenticazione Trasporto*)
protocol:GOVWAY-117	Rientrano in questa casistica gli errori avenuti durante la fase di autenticazione di una erogazione, se vengono rilevate credenziali non corrette (sezione *Autenticazione Trasporto*)
protocol:GOVWAY-1350	Rientrano in questa casistica eventuali errori generici avenuti durante la fase di autorizzazione di una erogazione (sezione *Autorizzazione*) o sicurezza del messaggio (sezione *Sicurezza a livello del messaggio*)
protocol:GOVWAY-1351	L'errore viene ritornato dal Gateway se viene rilevato che il messaggio presenta al suo interno un mittente differente da quello identificato dalle credenziali (configurazione non documentata)
protocol:GOVWAY-1352	Rientrano in questa casistica gli errori avenuti durante la fase di autorizzazione di una erogazione, quando la richieste non viene autorizzata (sezione *Autorizzazione*)
protocol:GOVWAY-[1353-1354]	L'errore viene ritornato dal Gateway se viene rilevato che la firma della busta, prevista dalla modalità utilizzata, non è rispettivamente valida o presente (configurazione non documentata)
protocol:GOVWAY-1355	L'errore viene ritornato dal Gateway se viene rilevato che la firma del messaggio non è valida (sezione *Sicurezza a livello del messaggio*)
protocol:GOVWAY-1356	L'errore viene ritornato dal Gateway se viene rilevato che il messaggio non è firmato (sezione *Sicurezza a livello del messaggio*)
protocol:GOVWAY-[1357-1360]	L'errore viene ritornato dal Gateway se viene rilevato che la firma degli allegati non sono valide o presenti (configurazione non documentata)
protocol:GOVWAY-1361	L'errore viene ritornato dal Gateway se viene rilevato che la cifratura del messaggio non è valida (sezione *Sicurezza a livello del messaggio*)
protocol:GOVWAY-1362	L'errore viene ritornato dal Gateway se viene rilevato che il messaggio non è cifrato (sezione *Sicurezza a livello del messaggio*)

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
protocol:GOVWAY-[1363-1364]	L'errore viene ritornato dal Gateway se viene rilevato che le cifrature degli allegati non sono valide o presenti (configurazione non documentata)
protocol:GOVWAY-1365	L'errore viene ritornato dal Gateway se viene rilevato che il messaggio non contiene l'attesa configurazione di sicurezza (sezione *Sicurezza a livello del messaggio*)
protocol:GOVWAY-1366	L'errore viene ritornato dal Gateway se non viene rilevato un token durante l'invocazione di una erogazione (sezione *Autenticazione Token*)
protocol:GOVWAY-1367	L'errore viene ritornato dal Gateway se viene rilevato un token non valido durante l'invocazione di una erogazione (sezione *Autenticazione Token*)
protocol:GOVWAY-1368	Rientrano in questa casistica gli errori avenuti durante la fase di autorizzazione di una erogazione, riguardanti la gestione di un token (sezione *Autorizzazione*)
protocol:GOVWAY-[1-6]	Rientrano in questa casistica gli errori generici avvenuti durante il processamento e la validazione di una richiesta di erogazione
protocol:GOVWAY-[51-60]	Gli errori che rientrano in questa casistica vengono generati durante la validazione della richiesta se sono presenti informazioni non valide per quanto concerne gli attributi *mustUnderstand* e *actor* di un header SOAP (es. busta egov nella modalità descritta in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[100-120]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul mittente (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[150-170]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul destinatario (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[200-205]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul profilo di collaborazione (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[250-265]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul servizio (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[300-315]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla correlazione asincrona (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[350-355]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla collaborazione (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[400-406]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sull'azione (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[450-455]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla correlazione asincrona per quanto riguarda l'azione (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[500-506]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sull'identificativo messaggio (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[550-556]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul riferimento messaggio (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[600-610]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sull'ora registrazione (es. busta egov in sezione *Profilo "SPCoop"*)

continues on next page

Tabella 9.2 – continua dalla pagina precedente

Codice	Descrizione
protocol:GOVWAY-[650-661]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla scandenza (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[700-717]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul filtro duplicati e sulla conferma della ricezione (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[750-766]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla consegna in ordine (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[800-817]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sul servizio applicativo
protocol:GOVWAY-[850-879]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sui riscontri (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[900-971]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla lista trasmissioni (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[1000-1035]	Errore rilevato durante la validazione della richiesta che riguarda informazioni sulla lista eccezioni (es. busta egov in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[1300-1329]	Errore rilevato durante la validazione del messaggio per quanto concerne la parte di SOAPFault previsto dal protocollo (es. busta egov errore in sezione *Profilo "SPCoop"*)
protocol:GOVWAY-[1400-1404]	Errore rilevato durante la validazione del messaggio per quanto concerne la parte di attachments previsto dal protocollo (es. busta egov con attachments, sezione *Profilo "SPCoop"*)
protocol:GOVWAY-2000	Errore generico rilevato durante la validazione del messaggio

CAPITOLO 10

Funzionalità Avanzate

10.1 Modalità Avanzata

L'interfaccia della govwayConsole, fin qui descritta, fa riferimento all'operatività nella *modalità standard*. La modalità standard prevede varie semplificazioni, sulle opzioni visualizzate nelle schermate, mirate al compimento delle operazioni di uso comune.

Nel caso si avesse la necessità di ricorrere a configurazioni più specifiche, non contemplate nella modalità standard, è possibile passare alla visualizzazione dell'interfaccia nella *Modalità Avanzata* utilizzando la voce omonima del menu a discesa che compare selezionando l'icona in alto a destra (nella testata della govwayConsole) come mostrato nella figura Fig. 10.1.

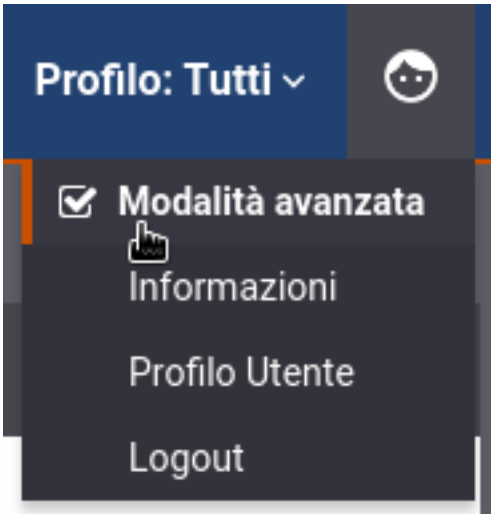


Fig. 10.1: Selezione Modalità Avanzata

Operando in modalità avanzata, in ciascuno dei contesti di configurazione già descritti in questo manuale, compariranno opzioni aggiuntive per le quali sono previsti valori di default nel caso della modalità standard.

Nella modalità avanzata sarà disponibile la funzionalità aggiuntiva *Elimina*, presente nel menu di Configurazione, che consente di utilizzare package di esportazione per l'eliminazione selettiva di entità dal registro.

Nota: Non tutte le funzionalità disponibili in modalità avanzata sono descritte nel presente manuale.

10.2 Configurazione manuale delle interfacce

Nel caso non si disponga del descrittore della API, è possibile in alternativa fornire manualmente la specifica delle interfacce. Dopo aver salvato la nuova API, senza aver fornito il descrittore delle interfacce, si procede individuando il nuovo elemento nella lista delle API presenti e cliccando sul collegamento presente nella colonna *Servizi*, nel caso SOAP, o *Risorse* nel caso REST.

Nel caso SOAP, si procede aggiungendo il nuovo servizio tramite il pulsante *Aggiungi*. Il form da compilare è quello mostrato nella figura seguente.

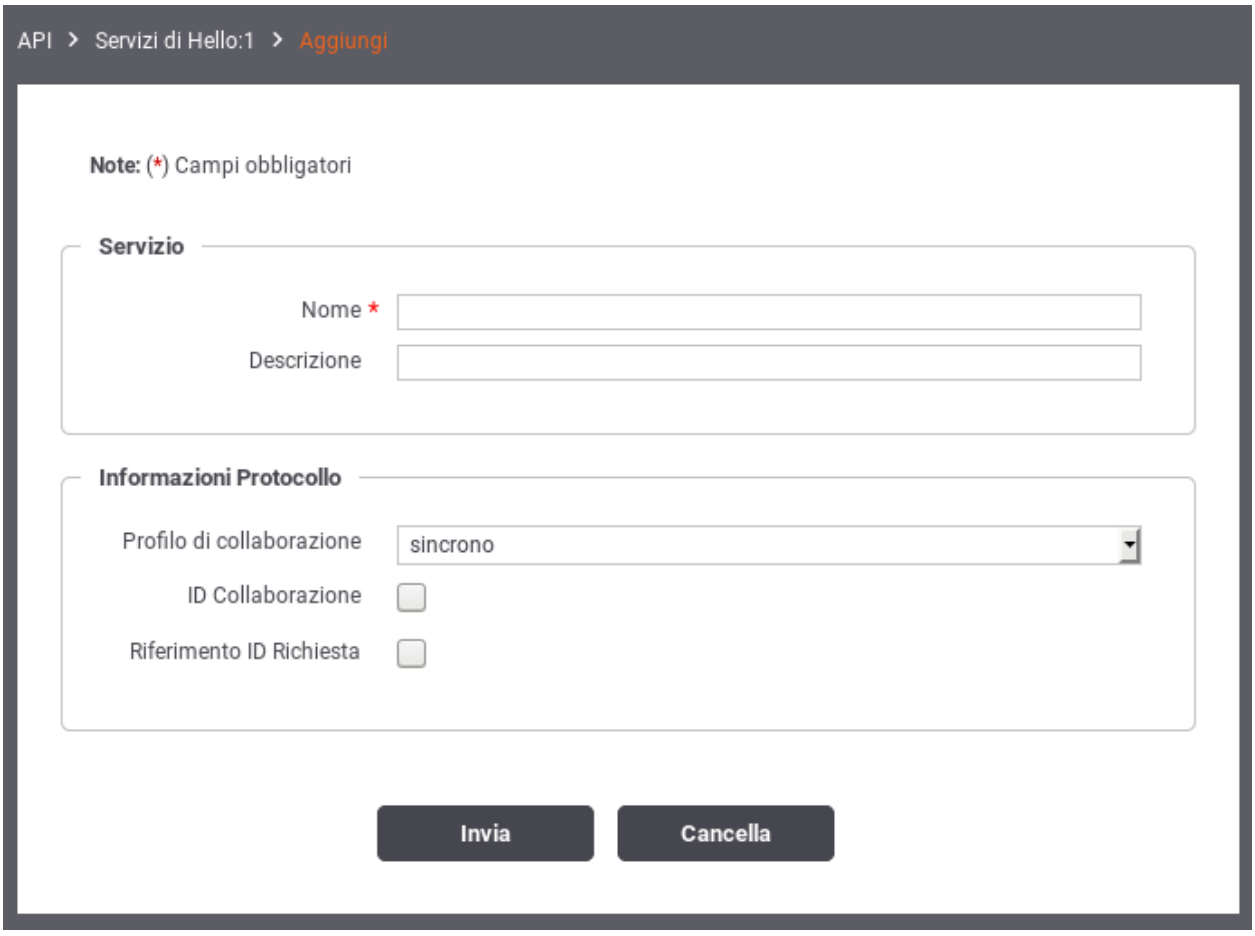


Fig. 10.2: Aggiunta di un servizio alla API SOAP

I dati da fornire sono i seguenti:

- *Nome* del servizio

- *Descrizione* del servizio
- *Profilo di collaborazione* del servizio, a scelta tra oneway e sincrono
- *ID Conversazione*. Flag per consentire di specificare nelle richieste un valore che identifica una conversazione.
- *Riferimento ID Richiesta*. Flag per consentire di specificare nelle richieste un identificativo relativo ad un messaggio precedente.

Al passo successivo, utilizzando il collegamento nella colonna *Azioni*, relativamente al servizio appena creato, si procede con l'aggiunta delle azioni. Il form da compilare è quello mostrato nella figura seguente.

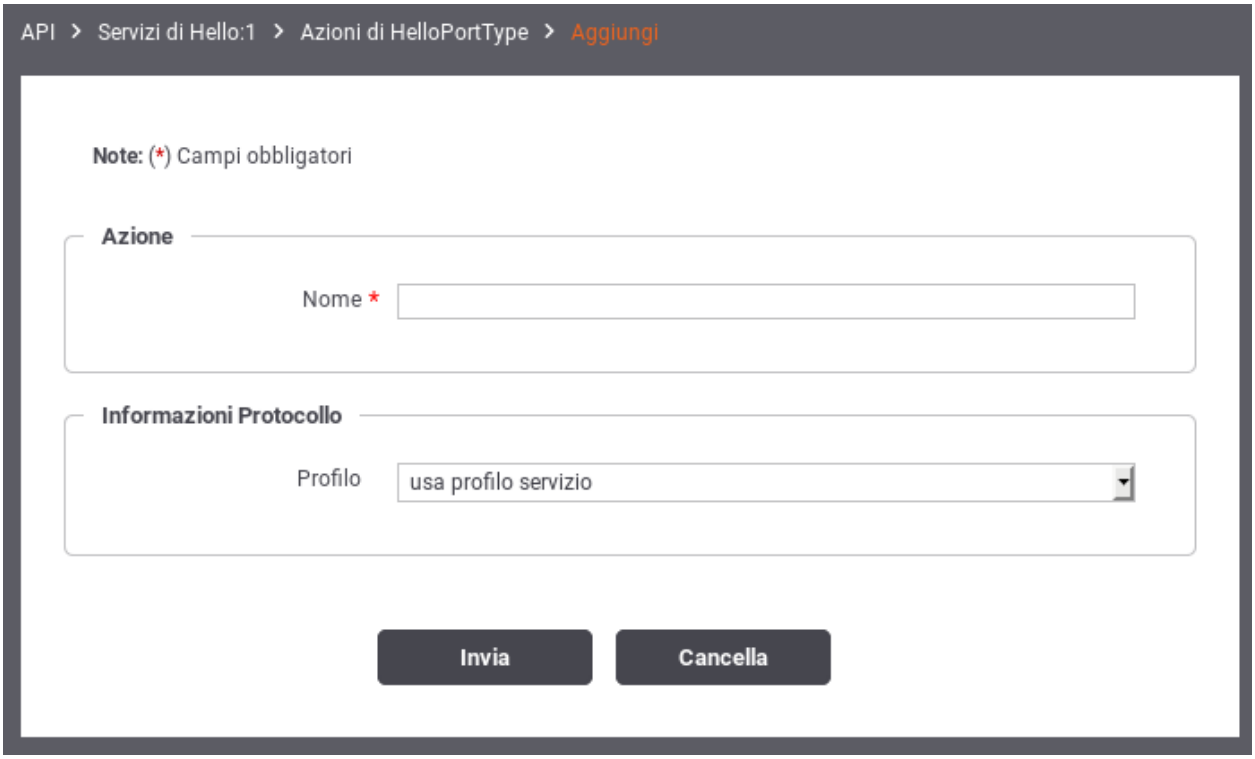


Fig. 10.3: Aggiunta di un'azione alla API SOAP

I dati da fornire sono i seguenti:

- *Nome* dell'azione
- *Profilo*. Si può scegliere se utilizzare le impostazioni già fornite a livello del servizio, oppure ridefinirle indicando nuovamente Profilo di collaborazione, ID Conversazione e Riferimento ID Richiesta.

Nel caso REST, si procede aggiungendo la nuova risorsa tramite il pulsante *Aggiungi*. Il form da compilare è quello mostrato nella figura seguente.

I dati da fornire sono i seguenti:

- *HTTP Method* relativo alla risorsa (GET, POST, DELETE, ecc.)
- *Path* della risorsa
- *Nome* della risorsa
- *Descrizione* della risorsa

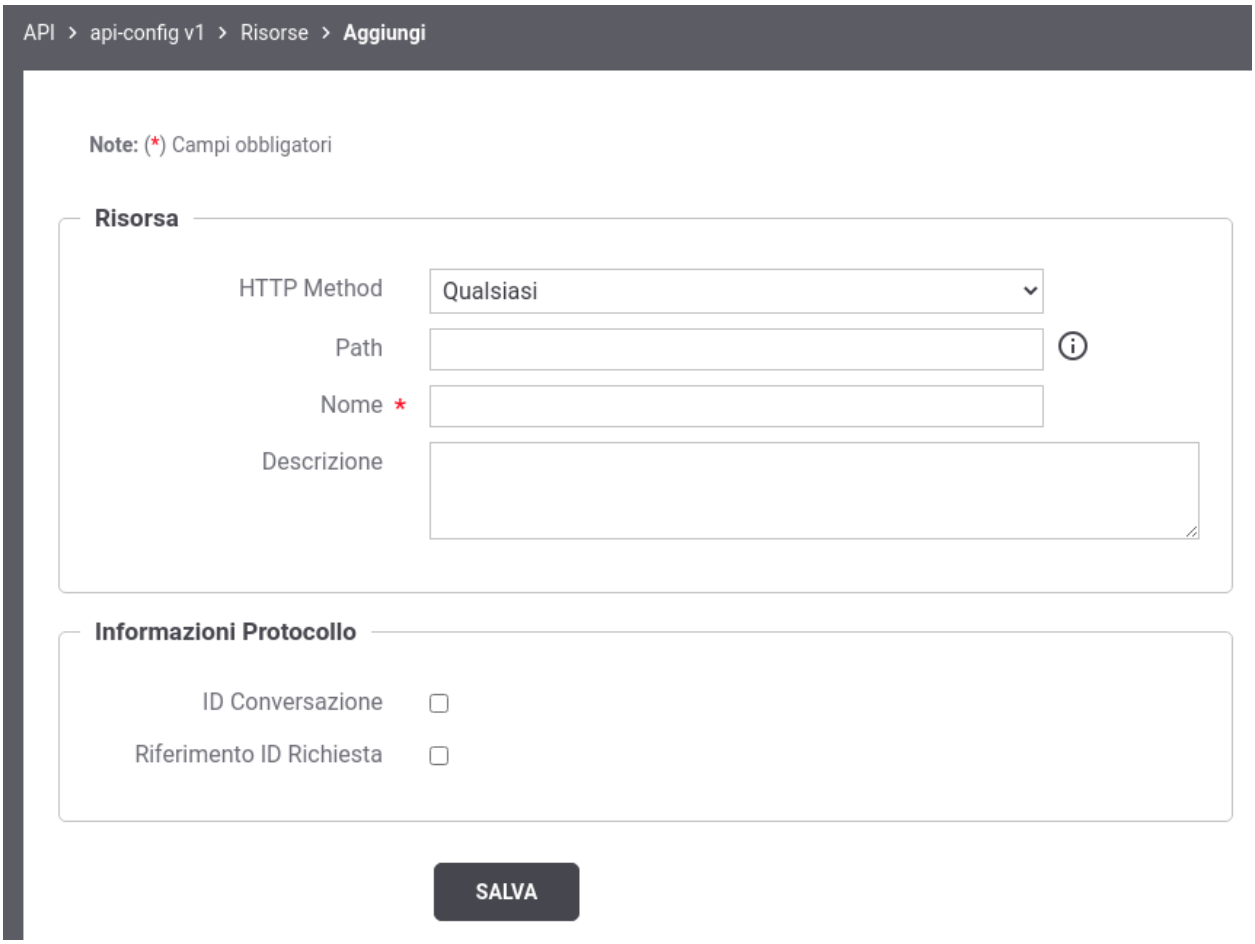


Fig. 10.4: Aggiunta di una risorsa alla API REST

- *ID Conversazione*. Flag per consentire di specificare nelle richieste un valore che identifica una conversazione.
- *Riferimento ID Richiesta*. Flag per consentire di specificare nelle richieste un identificativo relativo ad un messaggio precedente.

10.3 Versionamento delle API e delle Erogazioni/Fruizioni

Su GovWay vi è una gestione del versionamento effettuato su due componenti:

- API
- Erogazione o Fruizione dell'API

Come descritto nella sezione *Versionamento delle API*, sulla singola erogazione/fruizione è possibile modificare la versione dell'API implementata solamente se ne esiste più di una versione. Questa modifica si riflette automaticamente anche sulla versione dell'erogazione/fruizione, e sull'url di invocazione, se non esiste già una erogazione/fruizione con la nuova versione.

Utilizzando la console in modalità avanzata (*Modalità Avanzata*) è invece possibile modificare puntualmente la versione dell'erogazione/fruizione e di conseguenza l'url di invocazione tramite il bottone "modifica" evidenziato nella figura Fig. 10.5.

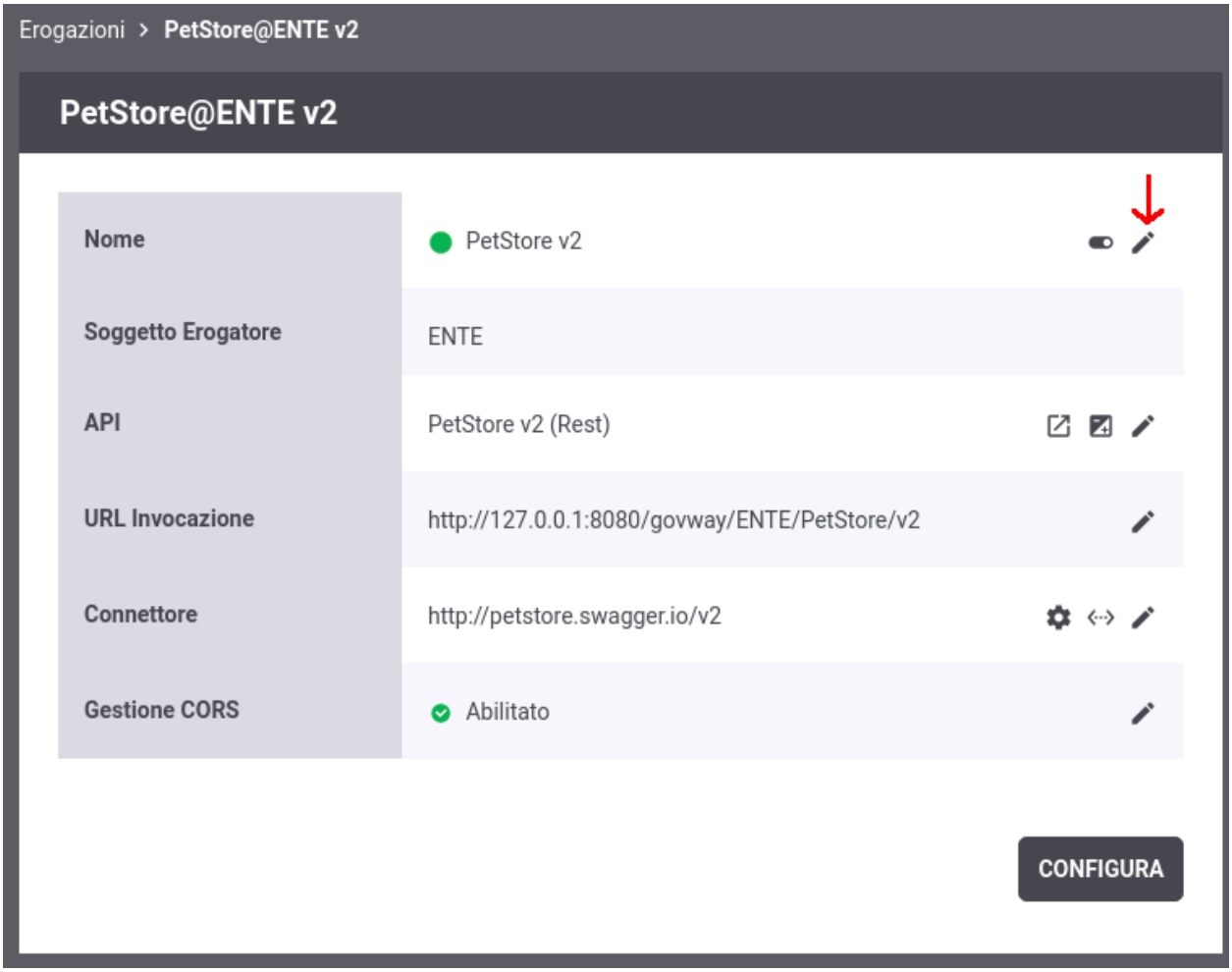


Fig. 10.5: Nuova Versione di una Erogazione

Accedendo alla modifica del nome dell'erogazione/fruizione con la console in modalità avanzata, è possibile modificare la versione (Fig. 10.6).

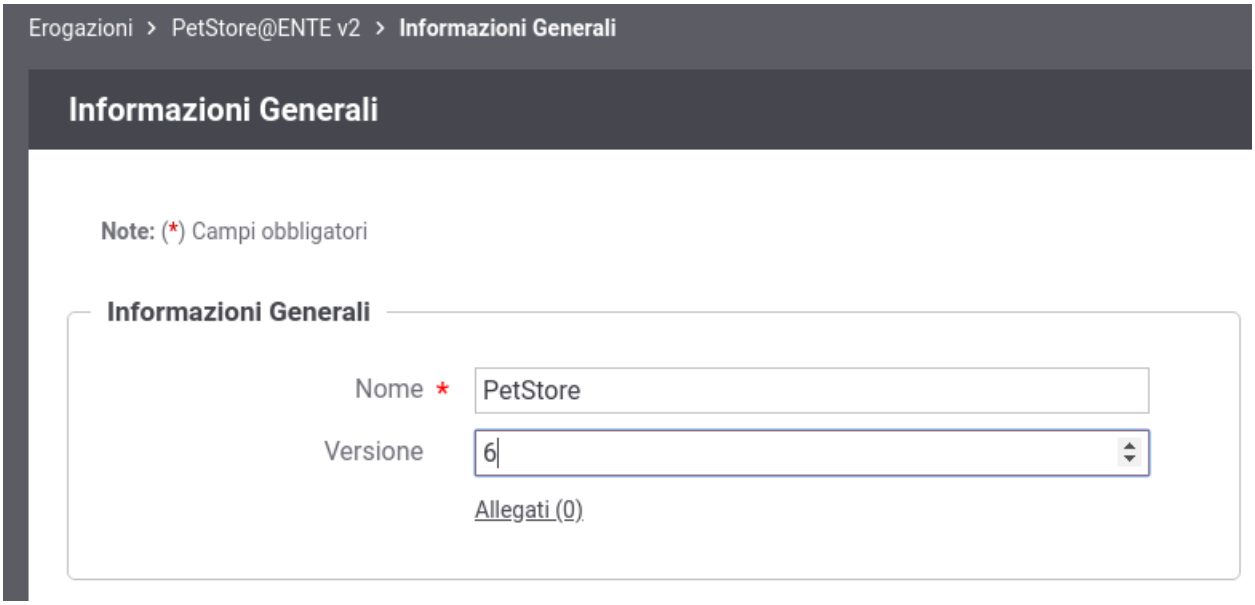


Fig. 10.6: Scelta di una nuova versione per una Erogazione

Effettuata la modifica l'erogazione possiederà una versione indipendente dalla versione dell'API implementata. L'url di invocazione riflette la versione dell'erogazione come evidenziato nella figura Fig. 10.7.

10.4 Modalità di identificazione dell'azione

Nel contesto dei servizi Soap, sia erogazioni che fruizioni, si ha la possibilità di selezionare una tra diverse opzioni che riguardano la modalità di identificazione dell'azione. Dopo aver acceduto la sezione *URL di Invocazione*, relativamente alla fruizione o erogazione, si può selezionare una tra le seguenti opzioni:

- *Contenuto* (Soap e Rest): il dato viene ricavato dal messaggio di richiesta utilizzando come criterio l'espressione XPath o JsonPath indicata nel campo *Pattern* sottostante.
- *Header HTTP* (Soap e Rest): il dato viene ricavato da un valore passato come Http Header. Il campo sottostante consente di specificare il nome di tale header.
- *Header di Integrazione* (Soap e Rest): il dato viene ricavato dall'header di integrazione fornito con il messaggio di richiesta. Per conoscere come gli applicativi client forniscono tale informazione vedere la sezione *Scambio di informazioni nella richiesta del client verso il gateway*.
- *Specifica di Interfaccia dell'API* (Soap e Rest): il dato viene ricavato in automatico sulla base delle informazioni fornite con la richiesta (messaggio e parametri) confrontandole con la descrizione dell'interfaccia dell'API.
- *Url di Invocazione* (Soap): il dato viene ricavato dinamicamente dalla url di invocazione utilizzando come criterio l'espressione regolare inserita nel campo *Espressione Regolare* sottostante (l'espressione deve avere un match con l'intera url).
- *SOAPAction* (Soap): Questa opzione consente di ricavare il dato dal campo *SOAPAction* presente nell'header di trasporto delle comunicazioni SOAP.

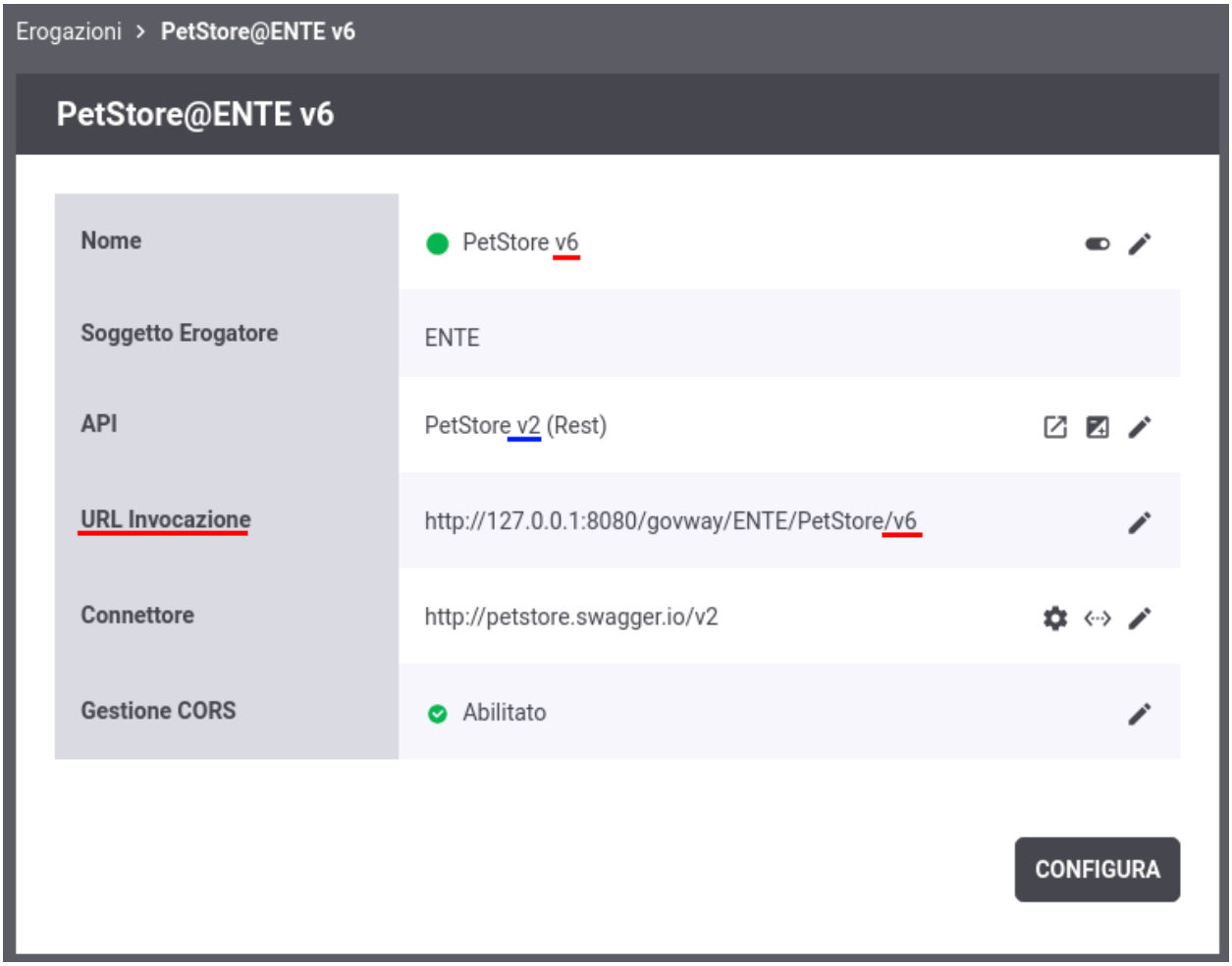


Fig. 10.7: Nuova versione dell'erogazione differente dalla versione dell'API

Attivando il flag *Identificazione tramite API*, in caso di fallimento dell'identificazione dell'azione nella modalità prevista al passo precedente, si tenterà di utilizzare la modalità «Specifica di Interfaccia dell'API» come seconda opzione.

Il campo *Azioni* illustra l'elenco delle azioni presenti per semplice comodità.

10.5 Multi-Tenant

I processi di configurazione, descritti in questo manuale, sono ottimizzati nell'ottica di mantenere sempre sottinteso il soggetto interno al dominio. In tal senso, le fruizioni e le erogazioni si intendono sempre in soggettiva riguardo un singolo soggetto interno amministrato dall'utente in sessione.

Multi-tenant è un'opzione che consente di estendere l'ambito delle configurazioni prodotte dalla govwayConsole a più di un soggetto interno al dominio. Tale opzione si attiva nella configurazione generale (sezione *Generale*).

Per gestire la compresenza di più soggetti interni al dominio, per la configurazione di erogazioni e fruizioni, è possibile scegliere quali soggetti interni rendere ammissibili (Fig. 10.8):

- *Fruizioni (Soggetto Erogatore)*
  - *Tutti*: indica che tutti i soggetti interni, censiti nel registro di GovWay, sono selezionabili come soggetto erogatore, in una fruizione.
  - *Escludi Soggetto Fruitore*: indica che tutti i soggetti interni, tranne il soggetto fruitore, sono selezionabili come soggetto erogatore, in una fruizione.
  - *Solo Soggetti Esterni*: indica che il soggetto erogatore di una fruizione deve essere un soggetto esterno.
- *Erogazioni (Soggetti Fruitori)*
  - *Tutti*: indica che tutti i soggetti interni, censiti nel registro di GovWay, sono selezionabili come soggetti fruitori, in una erogazione.
  - *Escludi Soggetto Erogatore*: indica che tutti i soggetti interni, tranne il soggetto erogatore, sono selezionabili come soggetti fruitori, in una erogazione.
  - *Solo Soggetti Esterni*: indica che i soggetti fruitori di una erogazione devono essere soggetti esterni.

Fig. 10.8: Elementi di configurazione della modalità multi-tenant

L'utente che ha l'opzione multi-tenant attiva, visualizza sulla testata un menu a discesa che consente di selezionare l'utente interno al dominio sul quale vuole operare (Fig. 10.9). Se viene selezionato un soggetto dalla lista, l'operatività

sulla console risulterà identica alla situazione con un unico soggetto interno. Selezionando l'opzione «Tutti» sarà richiesto nei singoli contesti di specificare il soggetto interno.

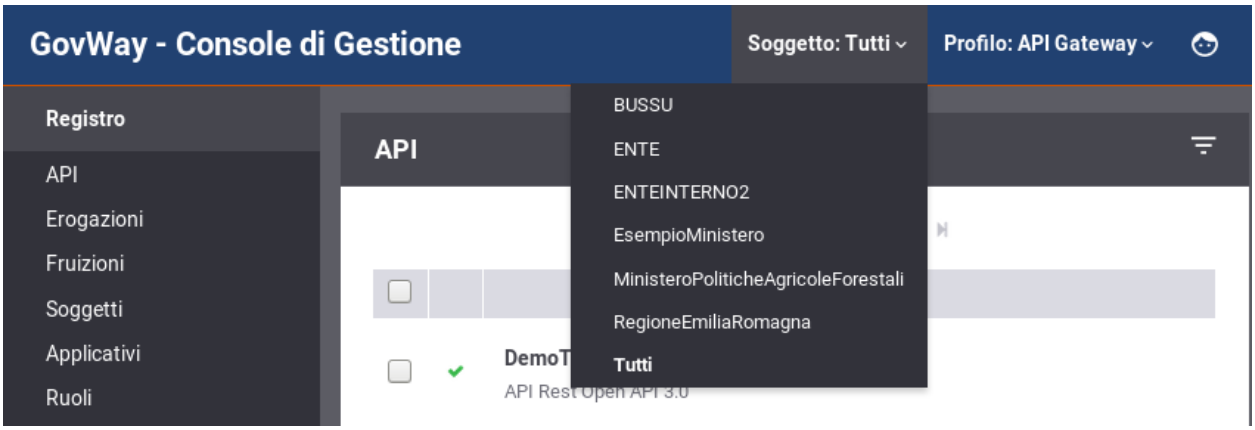


Fig. 10.9: Selezione del soggetto operativo in modalità multi-tenant

10.6 Header di Integrazione

In base alle configurazioni prodotte per i servizi, è previsto in diverse situazioni che gli applicativi scambino dei dati con il gateway.

Nel caso degli applicativi server lo scopo è quello di ricevere dal gateway i metadati che riguardano la richiesta gestita.

Per gli applicativi client tale scambio si rende necessario al fine di fornire al gateway specifici parametri necessari a elaborare la richiesta.

Per consentire lo scambio di tali informazioni, funzionali all'integrazione tra applicativi e gateway, sono previste alcune strutture dati, che indichiamo di seguito con il termine *Header di Integrazione*, che possono essere trasmesse in differenti modalità:

- *Trasporto*: le informazioni sono contenute nell'header di trasporto
- *Url Based*: le informazioni sono incapsulate nella url
- *SOAP*: le informazioni sono incluse in uno specifico header SOAP proprietario di GovWay
- *WS-Addressing*: le informazioni sono incluse in un header SOAP secondo il formato standard WS-Addressing

Nel seguito descriviamo le strutture degli header di integrazione attive per default con l'installazione del prodotto. Tali strutture variano in funzione del ruolo dell'applicativo. Per l'applicativo client è possibile fornire informazioni al gateway tramite le modalità *Trasporto* e *Url Based*. L'applicativo server, invece, riceve le informazioni dal gateway solamente tramite la modalità *Trasporto*.

10.6.1 Scambio di informazioni nella richiesta inoltrata dal gateway al server

Le informazioni fornite dal gateway all'applicativo erogatore, sia per quanto concerne fruizioni che per erogazioni, sono riassunte nella Tabella 10.1.

Tabella 10.1: Scambio di informazioni nella richiesta inoltrata dal gateway al server

Nome Header Trasporto	Descrizione
GovWay-Message-ID	Identificativo del messaggio assegnato da GovWay
GovWay-Relates-To	Identificativo del messaggio riferito
GovWay-Conversation-ID	Identificativo della conversazione
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

Inoltre, solamente per quanto concerne le erogazioni, all'applicativo interno al dominio vengono inoltrate ulteriori meta-informazioni riguardanti la transazione gestita sul gateway descritte nella Tabella 10.2.

Tabella 10.2: Scambio di informazioni nella richiesta inoltrata dal gateway al server per una Erogazione

Header	Descrizione
GovWay-Sender-Type	Codice che identifica il tipo del mittente
GovWay-Sender	Identificativo del mittente
GovWay-Provider-Type	Codice che identifica il tipo del destinatario
GovWay-Provider	Identificativo del destinatario
GovWay-Service-Type	Codice che identifica il tipo del servizio
GovWay-Service	Identificativo del servizio
GovWay-Service-Version	Progressivo di versione del servizio
GovWay-Action	Identificativo dell'azione
GovWay-Application-Message-ID	Identificativo del messaggio assegnato dall'applicativo
GovWay-Application	Identificativo dell'applicativo

10.6.2 Informazioni restituite dal gateway nella risposta all'applicativo client

Le informazioni restituite dal gateway all'applicativo client, sia per per le fruizioni che per le erogazioni, sono riassunte nella Tabella 10.3.

Tabella 10.3: Header restituiti dal gateway nella risposta all'applicativo client

Nome Header Trasporto	Descrizione
GovWay-Message-ID	Identificativo del messaggio assegnato da GovWay
GovWay-Relates-To	Identificativo del messaggio riferito
GovWay-Conversation-ID	Identificativo della conversazione
GovWay-Application-Message-ID	Identificativo del messaggio assegnato dall'applicativo (solo nel caso di Fruizione)
GovWay-Transaction-ID	Identificativo della transazione assegnato da GovWay

All'applicativo client vengono inoltre restituiti ulteriori header http informativi se l'applicativo erogatore non è disponibile o se sono stati attivati meccanismi di Rate Limiting (sezione *Rate Limiting*).

Tabella 10.4: Ulteriori header restituiti dal gateway nella risposta all'applicativo client

Nome Header Trasporto	Descrizione	Motivazione
Retry-After	Indica al client il numero di secondi dopo i quali ripresentarsi poichè il servizio contattato non è al momento disponibile.	Le principali cause della generazione di tale header sono imputabili alla non raggiungibilità un applicativo erogatore, alla violazione di politiche di RateLimiting o a quando un servizio è temporaneamente disabilitato
X-RateLimit-Limit	Indica il numero massimo di richieste effettuabili	Rate-Limiting attivato con policy di tipo "NumeroRichieste-ControlloRealtime*" (sezione *Rate Limiting*)
X-RateLimit-Remaining	Numero di richieste rimanenti prima del prossimo reset	Rate-Limiting attivato con policy di tipo "NumeroRichieste-ControlloRealtime*" (sezione *Rate Limiting*)
X-RateLimit-Reset	Numero di secondi mancante al prossimo reset	Rate-Limiting attivato con policy di tipo "NumeroRichieste-ControlloRealtime*" (sezione *Rate Limiting*)
GovWay-RateLimit-ConcurrentRequest-Limit	Indica il numero massimo di richieste concorrenti inviabili	Rate-Limiting attivato con policy di tipo "NumeroRichieste-RichiesteSimultanee" (sezione *Rate Limiting*)
GovWay-RateLimit-ConcurrentRequest-Remaining	Indica il numero massimo di richieste concorrenti ancora inviabili	Rate-Limiting attivato con policy di tipo "NumeroRichieste-RichiesteSimultanee" (sezione *Rate Limiting*)
GovWay-RateLimit-BandwithQuota-Limit	Indica la massima banda occupabile	Rate-Limiting attivato con policy di tipo "OccupazioneBanda-*" (sezione *Rate Limiting*)
GovWay-RateLimit-BandwithQuota-Remaining	Indica la banda ancora occupabile prima del prossimo reset	Rate-Limiting attivato con policy di tipo "OccupazioneBanda-*" (sezione *Rate Limiting*)
GovWay-RateLimit-BandwithQuota-Reset	Numero di secondi mancante al prossimo reset	Rate-Limiting attivato con policy di tipo "OccupazioneBanda-*" (sezione *Rate Limiting*)
GovWay-RateLimit-AvgTimeResponse-Limit	Tempo medio di risposta atteso	Rate-Limiting attivato con policy di tipo "TempoMedioRisposta-*" (sezione *Rate Limiting*)
GovWay-RateLimit-AvgTimeResponse-Reset	Numero di secondi mancante al prossimo reset	Rate-Limiting attivato con policy di tipo "TempoMedioRisposta-*" (sezione *Rate Limiting*)
GovWay-RateLimit-TimeResponseQuota-Limit	Tempo complessivo di risposta occupabile	Policy creata con risorsa di tipo "TempoComplessivioRisposta" (sezione *Registro Policy*)
GovWay-RateLimit-TimeResponseQuota-Remaining	Tempo di risposta ancora occupabile prima del prossimo reset	Policy creata con risorsa di tipo "TempoComplessivioRisposta" (sezione *Registro Policy*)
GovWay-RateLimit-TimeResponseQuota-Reset	Numero di secondi mancante al prossimo reset	Policy creata con risorsa di tipo "TempoComplessivioRisposta" (sezione *Registro Policy*)

continues on next page

Tabella 10.4 – continua dalla pagina precedente

Nome Header Trasporto	Descrizione	Motivazione
GovWay-RateLimit-RequestSuccessful-Limit, GovWay-RateLimit-RequestFailed-Limit, GovWay-RateLimit-Fault-Limit	Indica il numero massimo di richieste effettuabili	Policy creata rispettivamente con risorsa di tipo "NumeroRichiesteCompletateConSuccesso", "NumeroRichiesteFallite" e "NumeroFaultApplicativi" (sezione *Registro Policy*)
GovWay-RateLimit-RequestSuccessful-Remaining, GovWay-RateLimit-RequestFailed-Remaining, GovWay-RateLimit-Fault-Remaining	Numero di richieste rimanenti prima del prossimo reset	Policy creata rispettivamente con risorsa di tipo "NumeroRichiesteCompletateConSuccesso", "NumeroRichiesteFallite" e "NumeroFaultApplicativi" (sezione *Registro Policy*)
GovWay-RateLimit-RequestSuccessful-Reset, GovWay-RateLimit-RequestFailed-Reset, GovWay-RateLimit-Fault-Reset	Numero di secondi mancante al prossimo reset	Policy creata rispettivamente con risorsa di tipo "NumeroRichiesteCompletateConSuccesso", "NumeroRichiesteFallite" e "NumeroFaultApplicativi" (sezione *Registro Policy*)

10.6.3 Scambio di informazioni nella richiesta del client verso il gateway

Le informazioni che possono essere fornite dal client al gateway sono riassunte nella tabella Tabella 10.5 e riguardano le modalità *Trasporto* e *Url Based* attive di default.

Tabella 10.5: Scambio di informazioni nella richiesta del client verso il gateway

Nome Header Trasporto	Nome Url Property	Descrizione
GovWay-Action	govway_action	Identificativo dell'azione invocata. Tale informazione deve essere fornita dal client se il servizio è stato configurato in modalità di identificazione dell'azione *input-based*. (Sezione *Modalità di identificazione dell'azione*)
GovWay-Relates-To	govway_relates_to	Identificativo di un precedente messaggio a cui la richiesta in essere si riferisce. (Sezione *Correlazione tra transazioni differenti*)
GovWay-Conversation-ID	govway_conversation_id	Identificativo di una conversazione a cui la richiesta in essere si riferisce (Sezione *Correlazione tra transazioni differenti*)

10.6.4 Altri header di Integrazione

Per attivare ulteriori header di integrazione è richiesto l'accesso alla govwayConsole in modalità *avanzata* (Sezione *Modalità Avanzata*).

Nota: Gli header di trasporto relativi alle funzionalità di Rate-Limiting e Service-Unavailable, descritti nella sezione *Informazioni restituite dal gateway nella risposta all'applicativo client*, vengono generati solamente nella modalità *Header HTTP*.

A partire dall'erogazione o fruizione di una API, accedendo alla sezione Configurazione, descritta nella sezione *Configurazione dell'API*, in modalità avanzata compare una sezione precedentemente non documentata denominata *Opzioni Avanzate*. All'interno di tale sezione è possibile agire sulla configurazione della voce *Metadati* nella sezione *Integrazione* per attivare gli header di integrazione desiderati :

Nota: Per ogni tipo di header di integrazione descritto di seguito è possibile indicare, tramite una voce di configurazione dedicata, se deve essere generato solamente nei messaggi inoltrati al dominio interno (richiesta inoltrata al server nelle erogazioni o risposta restituita al client nelle fruizioni) o anche verso il dominio esterno.

- *Header HTTP*: vengono generati gli header di trasporto descritti nelle precedenti sezioni.
- *Parametri della Url*: le informazioni precedentemente descritte vengono aggiunte alla url tramite i parametri descritti nella Tabella 10.6.

Tabella 10.6: Informazioni generate dal gateway nella url della richiesta inoltrata al server

Nome Query URL Parameter	Descrizione
govway_message_id	Identificativo del messaggio assegnato da GovWay
govway_relates_to	Identificativo del messaggio riferito
govway_conversation_id	Identificativo della conversazione
govway_transaction_id	Identificativo della transazione assegnato da GovWay
govway_sender_type	Codice che identifica il tipo del mittente
govway_sender	Identificativo del mittente
govway_provider_type	Codice che identifica il tipo del destinatario
govway_provider	Identificativo del destinatario
govway_service_type	Codice che identifica il tipo del servizio
govway_service	Identificativo del servizio
govway_service_version	Progressivo di versione del servizio
govway_action	Identificativo dell'azione
govway_application_message_id	Identificativo del messaggio assegnato dall'applicativo
govway_application	Identificativo dell'applicativo

- *Header SOAP GovWay*: le informazioni precedentemente descritte vengono incluse come attributi in uno specifico header SOAP proprietario di GovWay che possiede il nome *integration* associato al namespace *http://govway.org/integration*. Di seguito un esempio di tale header:

```
<gw:integration
        ...
        transactionId="a2c6fd66-ec0b-407c-8a21-25b4920e7c73"
        SOAP_ENV:actor="http://govway.org/integration"
        SOAP_ENV:mustUnderstand="0"
        xmlns:SOAP_ENV="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:gw="http://govway.org/integration"/>
```

Nella tabella Tabella 10.7 vengono descritti i nome degli attributi.

Tabella 10.7: Informazioni generate dal gateway nell'header soap proprietario di GovWay

Nome Attributo	Descrizione
messageId	Identificativo del messaggio assegnato da GovWay
relatesTo	Identificativo del messaggio riferito
conversationId	Identificativo della conversazione
transactionId	Identificativo della transazione assegnato da GovWay
senderType	Codice che identifica il tipo del mittente
sender	Identificativo del mittente
providerType	Codice che identifica il tipo del destinatario
provider	Identificativo del destinatario
serviceType	Codice che identifica il tipo del servizio
service	Identificativo del servizio
serviceVersion	Progressivo di versione del servizio
action	Identificativo dell'azione
applicationMessageId	Identificativo del messaggio assegnato dall'applicativo
application	Identificativo dell'applicativo

Nota: Utilizzabile solamente con API di tipologia SOAP

- *WS-Addressing*: all'interno del messaggio Soap vengono generati gli header *To*, *From*, *Action*, *MessageID* e *RelatesTo* associati al namespace *http://www.w3.org/2005/08/addressing*. I valori utilizzati per i vari header sono i seguenti:
  - *To*

```
http://<providerType>_<provider>.govway.org/services/<serviceType>_
↪<service>/<serviceVersion>
```

  - *From*

```
http://[<application>.]<senderType>_<sender>.govway.org
```

  - *Action*

```
http://<providerType>_<provider>.govway.org/services/<serviceType>_
↪<service>/<serviceVersion>/<action>
```

  - *MessageID* di Protocollo, ritornato in una risposta di una fruizione o inserito nella consegna della richiesta di una erogazione

```
uuid:<messageId>
```

  - *MessageID* di Integrazione, atteso nella richiesta inviata dal client in una fruizione o nella risposta ritornata dal backend in una erogazione. Viene utilizzato ad es. per la funzionalità di correlazione applicativa

```
uuid:<applicationMessageId>
```

  - *RelatesTo*

```
uuid:<relatesTo>
```

Nota: Utilizzabile solamente con API di tipologia SOAP

- *Template*: consente di definire tramite un template freemaker o velocity come le informazioni siano inserite nel messaggio di richiesta, di risposta o in entrambi. Il tipo di template (freemarker/velocity) e il path del file template possono essere specifici per singole API indicandoli nelle proprietà "integrazione.template.richiesta/risposta.tipo" e "integrazione.template.richiesta/risposta.file". In alternativa è possibile definire il tipo e il file template a livello globale agendo sul file locale di configurazione *govway_local.properties* tramite la definizione delle proprietà "org.openspcoop2.pdd.integrazione.template.<pd/pa>.<request/response>.tipo" e "org.openspcoop2.pdd.integrazione.template.<pd/pa>.<request/response>.file".
- *Header HTTP di Autenticazione*: consente di generare Header HTTP utilizzabili dal backend per autenticare l'API Gateway. I nomi degli header generati ed i loro valori possono essere specifici per singole API indicandoli nelle proprietà "integrazione.autenticazione.headers" e "integrazione.autenticazione.header.<NOME_HEADER>". In alternativa è possibile definire gli header a livello globale agendo sul file locale di configurazione *govway_local.properties* tramite la definizione delle proprietà "org.openspcoop2.pdd.integrazione.autenticazione.<pd/pa>.request.headers" e "org.openspcoop2.pdd.integrazione.autenticazione.<pd/pa>.request.header.<NOME_HEADER>".
- *OpenSPCoop 2.x* o *OpenSPCoop2 1.x*: sono disponibili header di integrazione compatibili con le versioni di OpenSPCoop 2.x e 1.x:
  - Header HTTP: le informazioni sono veicolate all'interno di header HTTP. È possibile indicare se i nomi degli header debbano possere o meno il prefisso "X-";
  - Parametri Url: le informazioni sono veicolate come parametri della url
  - Header SOAP: le informazioni sono incluse in uno specifico header SOAP proprietario di OpenSPCoop 2.x o 1.x

10.7 Connettori

I connettori rappresentano le entità di configurazione che consentono a GovWay di indirizzare le comunicazioni verso gli attori dei flussi di erogazione/fruizione gestiti. Nel nostro contesto possiamo distinguere due tipologie di comunicazioni:

- *GovWay —> Applicativo Esterno*, nel caso di fruizioni
- *GovWay —> Applicativo Interno*, nel caso di erogazioni

I connettori di GovWay permettono di configurare differenti aspetti della comunicazione http:

- *Autenticazione http*: tale funzionalità permette di impostare delle credenziali http basic (username e password).
- *Autenticazione token*: tale funzionalità permette di inoltrate un Bearer Token.
- *Autenticazione https*: se l'utente lo desidera può personalizzare tutti gli aspetti che riguardano una comunicazione sicura su https.
- *Proxy*: è possibile configurare un proxy http che media la comunicazione.
- *Ridefinisci Tempi Risposta*: permette di ridefinire i tempi di risposta che sono stati configurati a livello generale, nell'ambito del controllo del traffico (vedi sezione *Tempi Risposta*).

Attivando la *modalità avanzata* dell'interfaccia saranno inoltre disponibili le seguenti opzioni:

- *Data Transfer Mode*: tramite questa configurazione è possibile indicare se la comunicazione deve avvenire in modalità transfer-encoding-chunked (streaming) o content length fisso.

- *Redirect*: tramite questa configurazione è possibile indicare se un eventuale redirect ritornato dal server contattato deve essere seguito o meno.
- *Debug*: è possibile abiltare un log verboso di tutta la comunicazione.

La govwayConsole, tramite l'interfaccia in modalità *avanzata*, consente anche di configurare le comunicazioni attraverso connettori non basati sul protocollo HTTP (o HTTPS). GovWay offre built-in i seguenti ulteriori connettori:

- *JMS*: connettore basato sul protocollo JMS
- *File*: connettore che permette di serializzare il messaggio di richiesta su FileSystem ed opzionalmente generare una risposta.
- *Null*: connettore per test. Si comporta come un servizio Oneway ricevendo richieste senza rispondere
- *NullEcho*: connettore per test. Si comporta come un servizio Sincrono rispondendo con un messaggio identico alla richiesta

Nel seguito vengono descritte alcune funzionalità specifiche dei connettori HTTP e HTTPS. Inoltre viene fornita una descrizione del connettore built-in JMS e File.

10.7.1 Autenticazione Http

Quando si configura l'autenticazione per un connettore è possibile scegliere la modalità di autenticazione HTTP-BASIC. Tale funzionalità permette di impostare delle credenziali (username e password) che verranno iniettate nella comunicazione http tramite header "Authorization" ("https://tools.ietf.org/html/rfc2617#section-2").

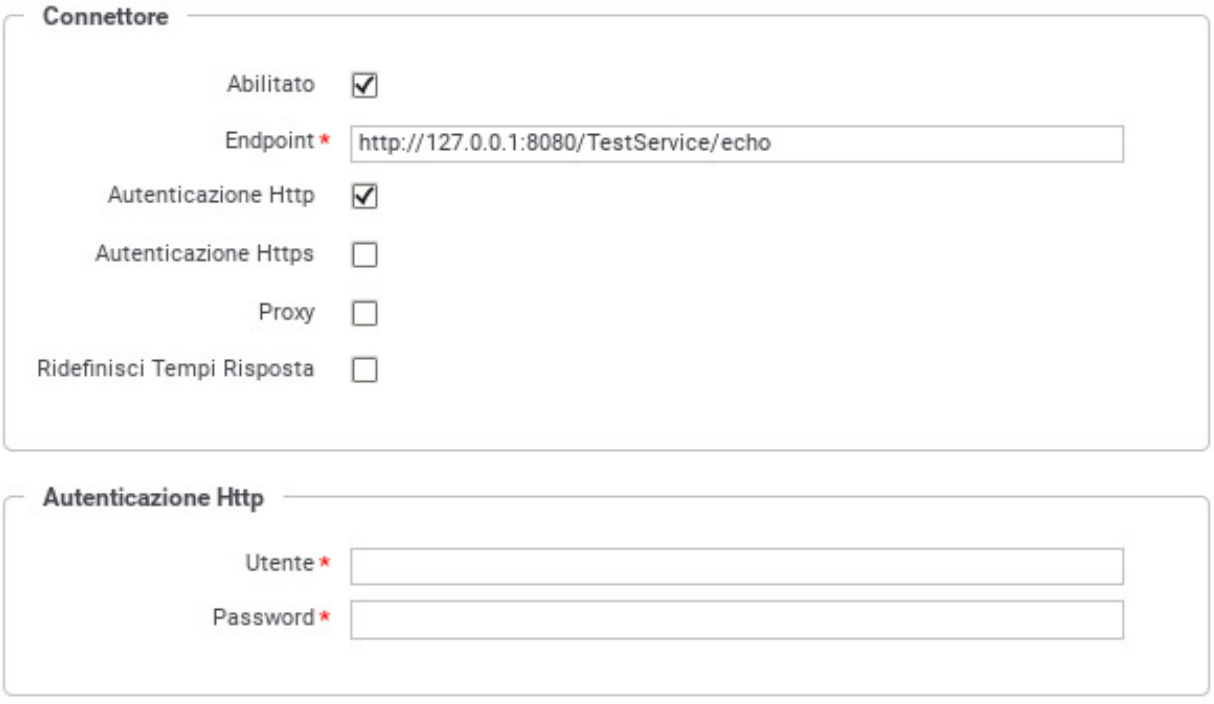


Fig. 10.10: Dati di configurazione di un'autenticazione Http

10.7.2 Autenticazione Token

Quando si configura l'autenticazione per un connettore è possibile scegliere la modalità di autenticazione per token. Tale funzionalità permette di iniettare un Token Bearer nella comunicazione http tramite la modalità definita all'interno della policy selezionata (es. tramite header "Authorization"). Per ulteriori dettagli su come registrare una policy di negoziazione del Bearer Token si rimanda alla sezione *Token Policy Negoziazione*.

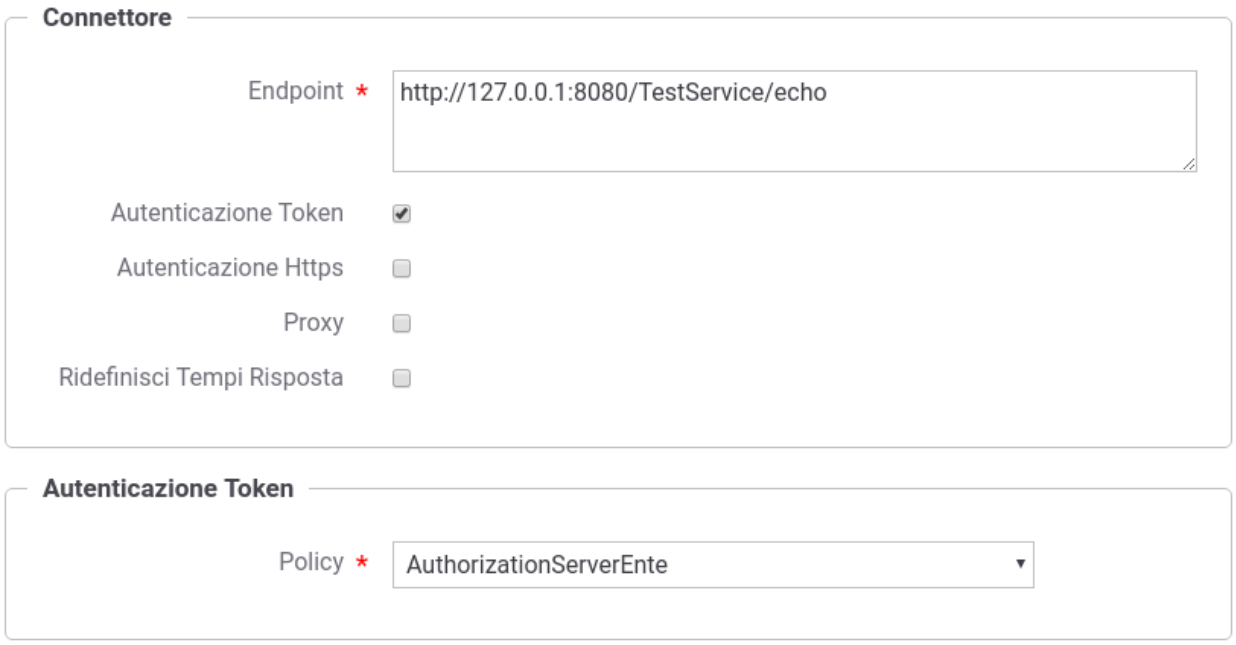


Fig. 10.11: Dati di configurazione di un'autenticazione Token

10.7.3 Autenticazione Https

Il connettore HTTPS permette di personalizzare i parametri SSL per ogni connessione che utilizza questo protocollo.

Il connettore HTTPS supporta:

- **Autenticazione Server**, è possibile definire le trusted keys e indicare se si desidera verificare l'hostname rispetto al certificato server contenuto nella sessione SSL.
- **Autenticazione Client**, è opzionale; se abilitata permette di definire il keystore contenente la chiave privata che si deve utilizzare durante la sessione SSL.

Facendo riferimento alla maschera raffigurata in Fig. 10.12 andiamo a descrivere il significato dei parametri:

- *Connettore*
  - **Url**: indirizzo endpoint del connettore
  - **Tipologia** (es. TLSv1.2): Tipo e versione del protocollo di trasporto. Sono selezionabili tutti i tipi supportati dalla versione della jvm utilizzata.
  - **Verifica Hostname** (true/false): Attiva la verifica in fase di autenticazione server della corrispondenza tra l'hostname indicato nella url e quello presente nel certificato server ritornato dal server (nel subject CN=hostname)

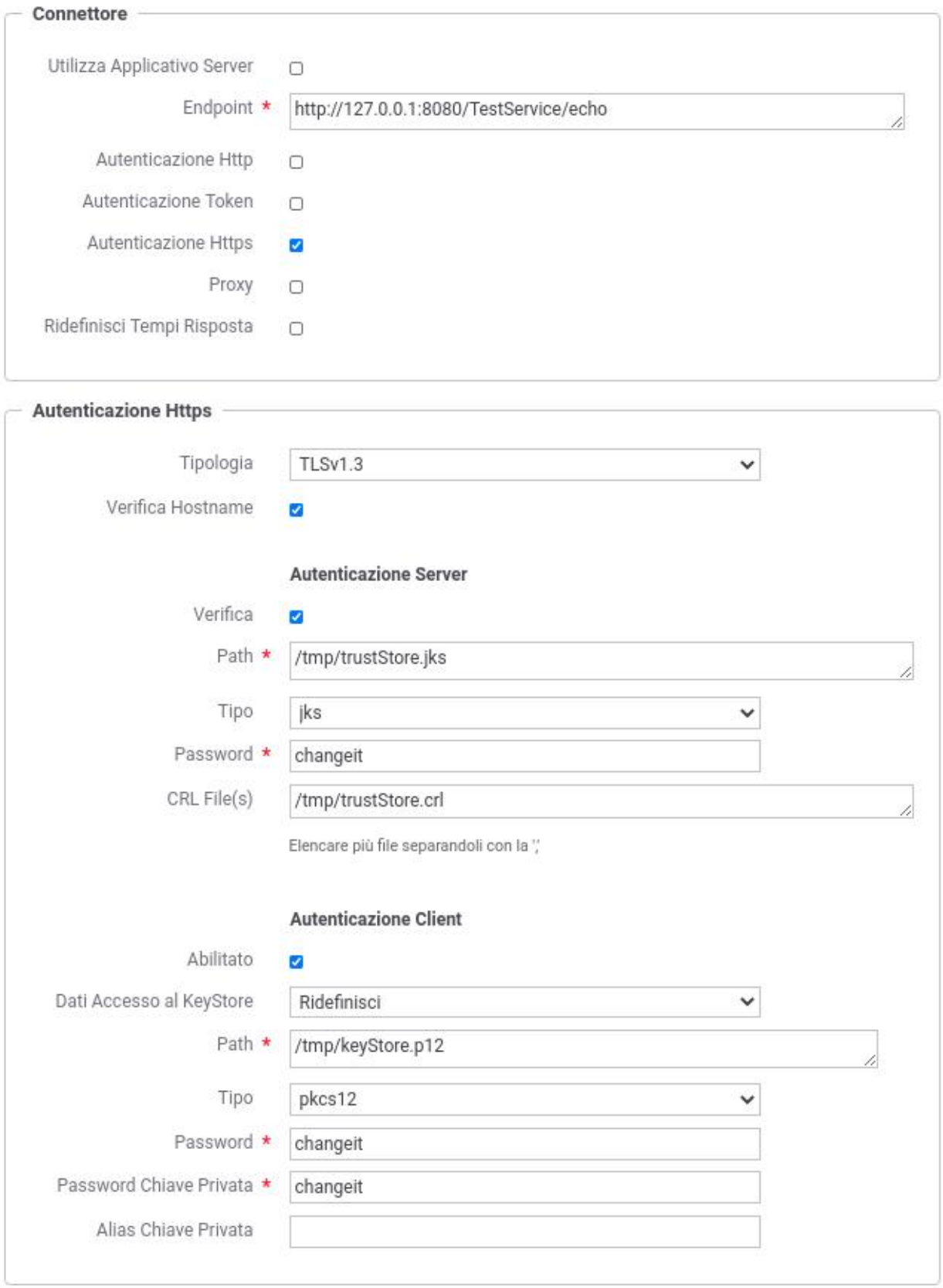


Fig. 10.12: Dati di configurazione di un'autenticazione Https

- *Autenticazione Server*

  I certificati server saranno validati tramite la configurazione indicata di seguito. Per accettare qualsiasi certificato restituito dal server è possibile disattivare la **Verifica**.

  - **Path**: Path dove è localizzato il truststore contenente i certificati server trusted.
  - **Tipo** (jks, pkcs12, jceks, bks, uber e gkr): Tipologia del TrustStore (default: jks)
  - **Password**: Password per l'accesso al TrustStore
  - **CRL File(s)**: Path dove è presente una CRL da utilizzare per validare i certificati server. L'indicazione di una CRL è opzionale e ne possono essere indicate più di una separando i path con la virgola.

- *Autenticazione Client (opzionale)*

  Abilitando la checkbox **Abilitato** è possibile configurare il certificato client che verrà inoltrato al server.

  - **Dati di Accesso al KeyStore** (usa valori del TrustStore, Ridefinisci): Consente di riutilizzare i medesimi riferimenti del TrustStore anche per il KeyStore o in alternativa ridefinirli.
  - **Tipo (solo se Dati di Accesso ridefiniti)** (jks, pkcs12, jceks, bks, uber e gkr): Tipologia del Keystore (default: jks)
  - **Password (solo se Dati di Accesso ridefiniti)**: Password per l'accesso al Keystore
  - **Password Chiave Privata**: Password per accedere alla chiave privata presente nel keystore.
  - **Alias Chiave Privata**: Alias che individua la chiave privata, presente nel keystore, da utilizzare. L'indicazione di un alias è opzionale e se non fornito viene utilizzata la prima chiave trovata.

10.7.4 Proxy

Funzionalità che consente di configurare un proxy http che media la comunicazione. Oltre ai classici parametri hostname e porta, è possibile anche indicare delle credenziali http basic (username e password) che verranno iniettate nella comunicazione http tramite header "Proxy-Authorization".

10.7.5 Tempi Risposta

Tramite questa sezione è possibile ridefinire i tempi di risposta che sono stati configurati a livello generale, nell'ambito del controllo del traffico (vedi sezione *Tempi Risposta*).

10.7.6 Configurazione Http Avanzata

Richiede accesso alla govwayConsole in modalità *avanzata*

Tramite questa sezione è possibile indicare sia quale modalità di comunicazione (streaming o meno) deve essere utilizzata, sia se deve avvenire una eventuale gestione dei redirect http.

Facendo riferimento alla maschera raffigurata in Fig. 10.14 andiamo a descrivere il significato dei parametri:

- *Data Transfer Mode* tramite questa configurazione è possibile indicare se la comunicazione deve avvenire in modalità transfer-encoding-chunked (streaming) o con content length fisso.

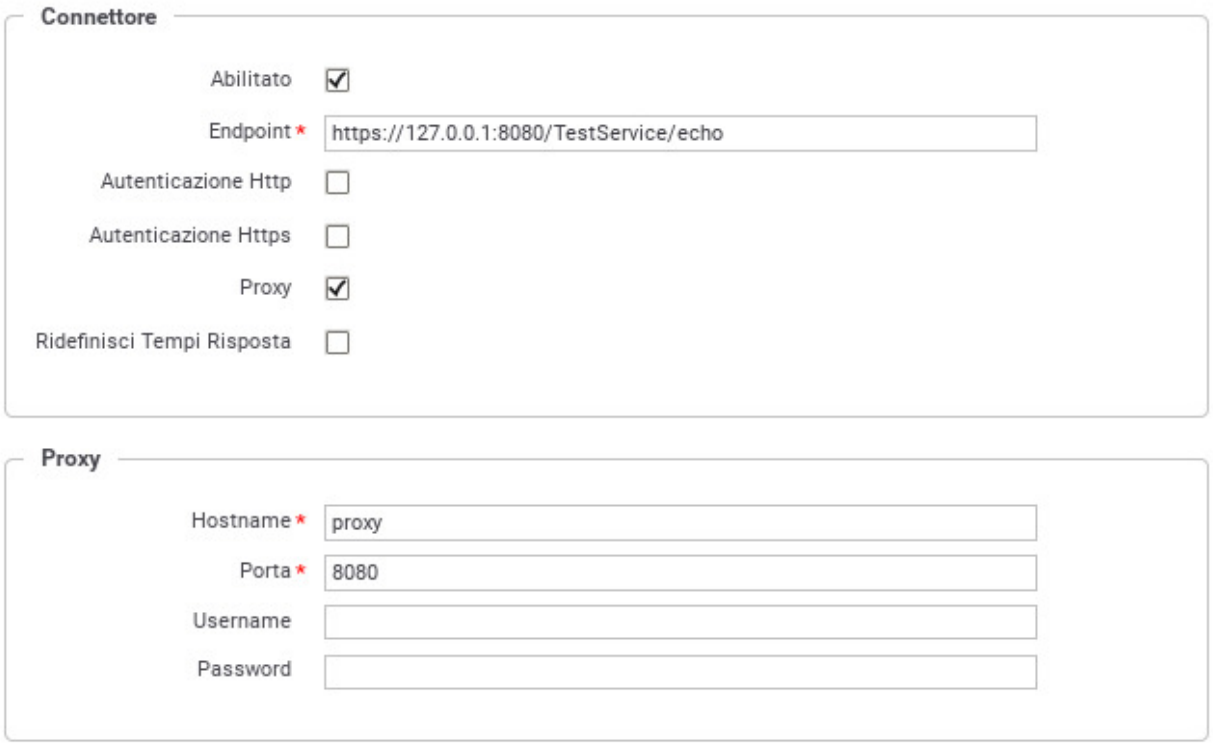


Fig. 10.13: Dati di configurazione di un Proxy Http

  - **Modalità Data Transfer** (default, content-length, transfer-encoding-chunked): indica il tipo di trasferimento dati; scegliendo la voce default verrà utilizzato il comportamento configurato a livello globale nel file govway.properties tramite le opzioni:
    - org.openspcoop2.pdd.connettori.inoltroBuste.httpTransferLength
    - org.openspcoop2.pdd.connettori.consegnaContenutiApplicativi.httpTransferLength
  - **Chunk Length (Bytes)** (presente solamente se la modalità è transfer-encoding-chunked): indica la dimensione in bytes di ogni singolo http chunk.

- *Redirect* tramite questa configurazione è possibile indicare se un eventuale redirect ritornato dal server contattato deve essere seguito o meno.
  - **Gestione Redirect** (default, abilitato, disabilitato): consente di personalizzare il comportamento sul singolo connettore; scegliendo la voce default verrà utilizzato il comportamento configurato a livello globale nel file govway.properties tramite le opzioni:
    - org.openspcoop2.pdd.connettori.inoltroBuste.followRedirects
    - org.openspcoop2.pdd.connettori.consegnaContenutiApplicativi.followRedirects
  - **Massimo Numero di Redirect** (presente solamente se la gestione redirect è abilitata): indica il massimo numero di redirect seguiti.

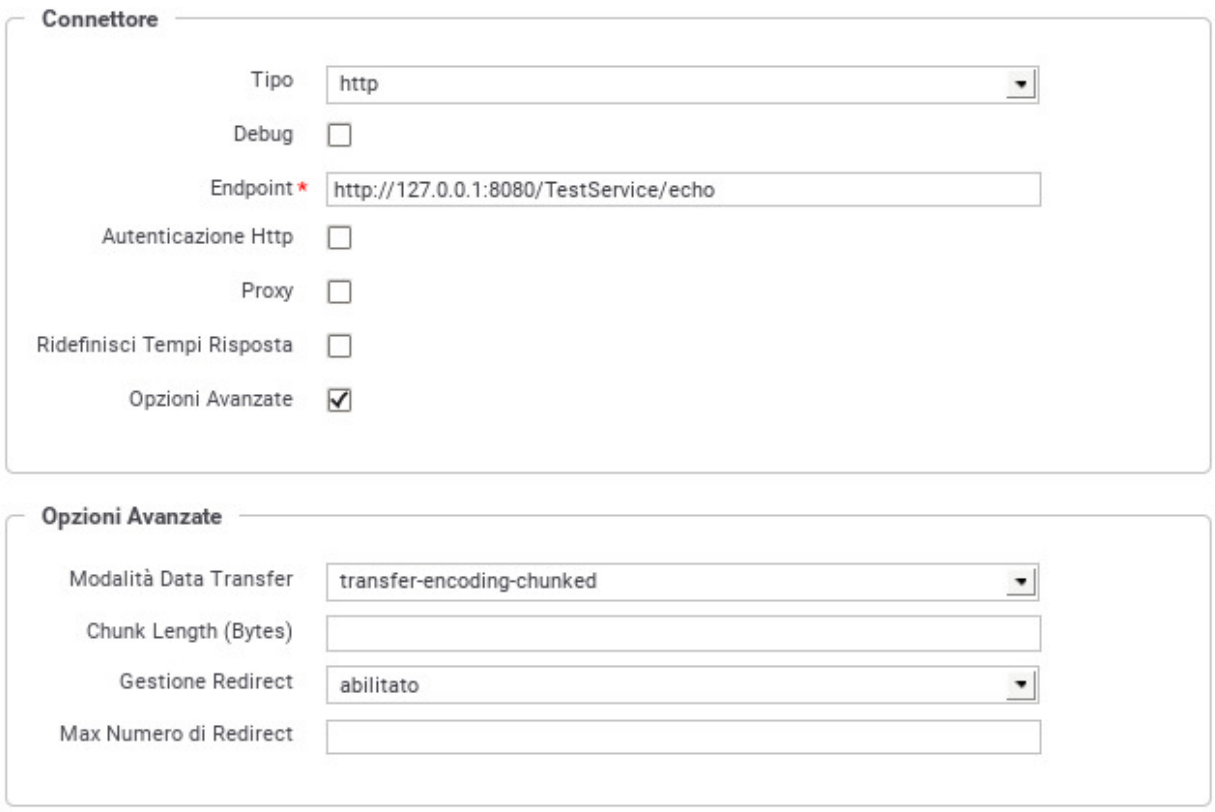


Fig. 10.14: Configurazione http avanzata

10.7.7 Debug

Richiede accesso alla govwayConsole in modalità *avanzata* (sezione *Modalità Avanzata*).

Se viene abilitato il debug, GovWay produce un log verboso di tutta la comunicazione nel file

- */var/log/govway/govway_connettori.log*

(assumendo che */var/log/govway* sia la directory di logging configurata)

Fig. 10.15: Debug

10.7.8 Connettore JMS

Il connettore JMS consente di configurare i parametri per abilitare la comunicazione tra GovWay e gli applicativi attraverso il protocollo JMS.

In Fig. 10.16 è mostrata la maschera di configurazione del connettore JMS.

In riferimento alla Fig. 10.16 descriviamo in dettaglio il significato dei campi per la configurazione:

- **Nome**: identificatore JNDI della risorsa queue/topic JMS
- **Tipo** (Queue/Topic): Si specifica se la risorsa JMS è di tipo queue o topic
- **Send As** (TextMessage/BytesMessage): Si sceglie la codifica del messaggio da inviare tramite broker JMS, tra TextMessage e BytesMessage.
- **Utente**: Username relativo alle credenziali per l'autenticazione e la negoziazione di una connessione sul Broker JMS
- **Password**: Password relativa alle credenziali per l'autenticazione e la negoziazione di una connessione sul Broker JMS
- **Connection Factory**: Identificatore della risorsa JNDI per la creazione di una connessione verso il broker JMS
- **Initial Context Factory**: Class Name per l'inizializzazione del server JNDI per la lookup della Connection Factory e della Coda

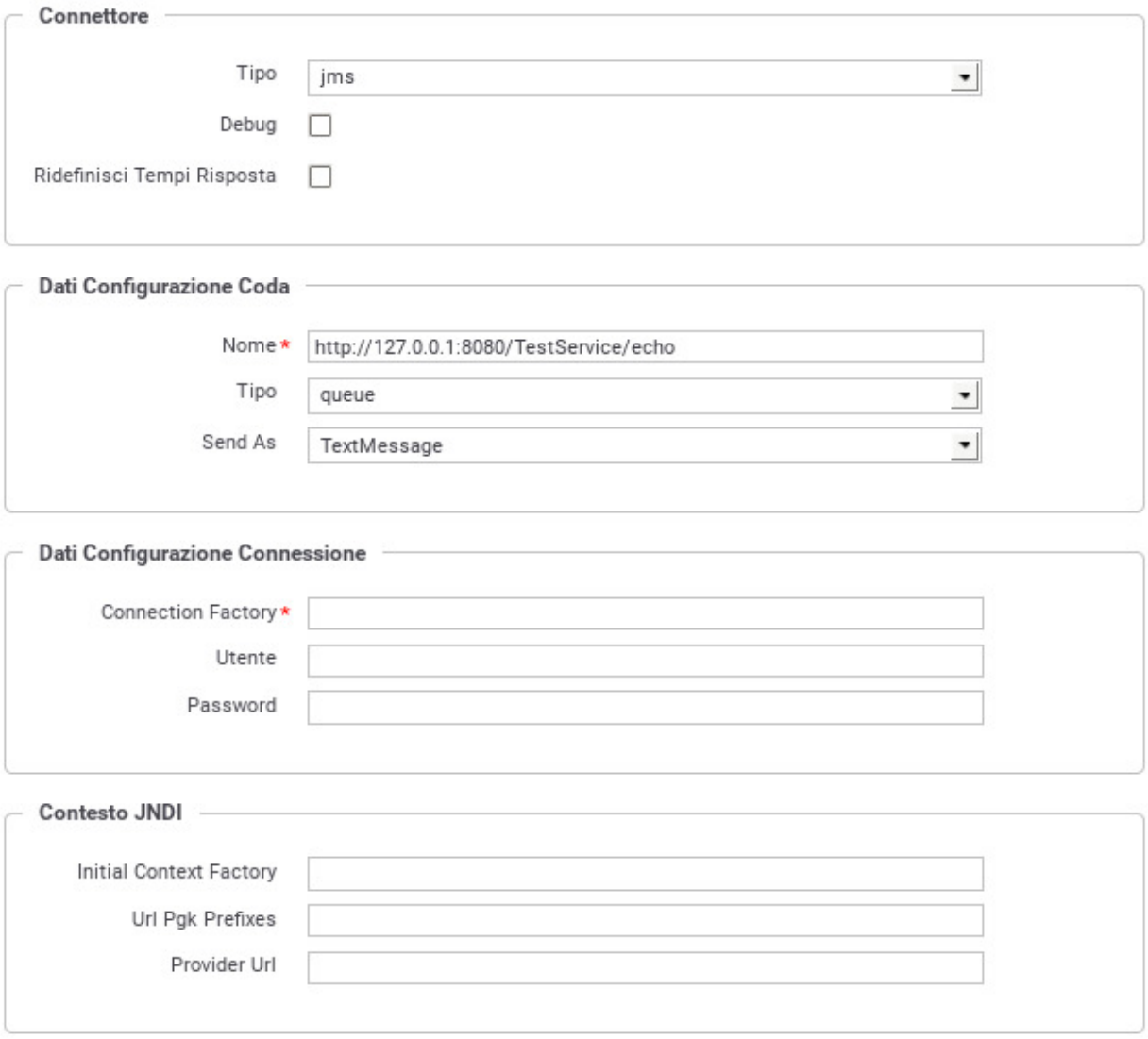


Fig. 10.16: Dati di configurazione di un connettore JMS

- **Url Pkg Prefixes**: Lista sperata da ":" per specificare i prefissi dei package da utilizzare per l'inizializzazione del Context JNDI
- **Provider Url**: Indirizzo che localizza il server JNDI

10.7.9 Connettore File

Il connettore permette di serializzare la richiesta su FileSystem ed opzionalmente di generare una risposta.

Il connettore File supporta:

- **Richiesta**, è possibile serializzare il messaggio di richiesta su file-system fornendo un path che può contenere anche parti dinamiche risolte a runtime da GovWay. È permesso anche abilitare l'eventuale sovrascrittura del file, se risulta già esistente, e la creazione automatica delle directory padre, se non esistono.
- **Risposta**, è opzionale; se abilitata permette di generare una risposta costruita utilizzando il contenuto di un file indirizzabile a sua volta tramite gli stessi meccanismi dinamici della richiesta. Il file contenente la risposta può essere eliminato una volta consumato (opzione configurabile). L'utente può inoltre indicare un tempo di attesa (ms) qualora il file non sia immediatamente disponibile.

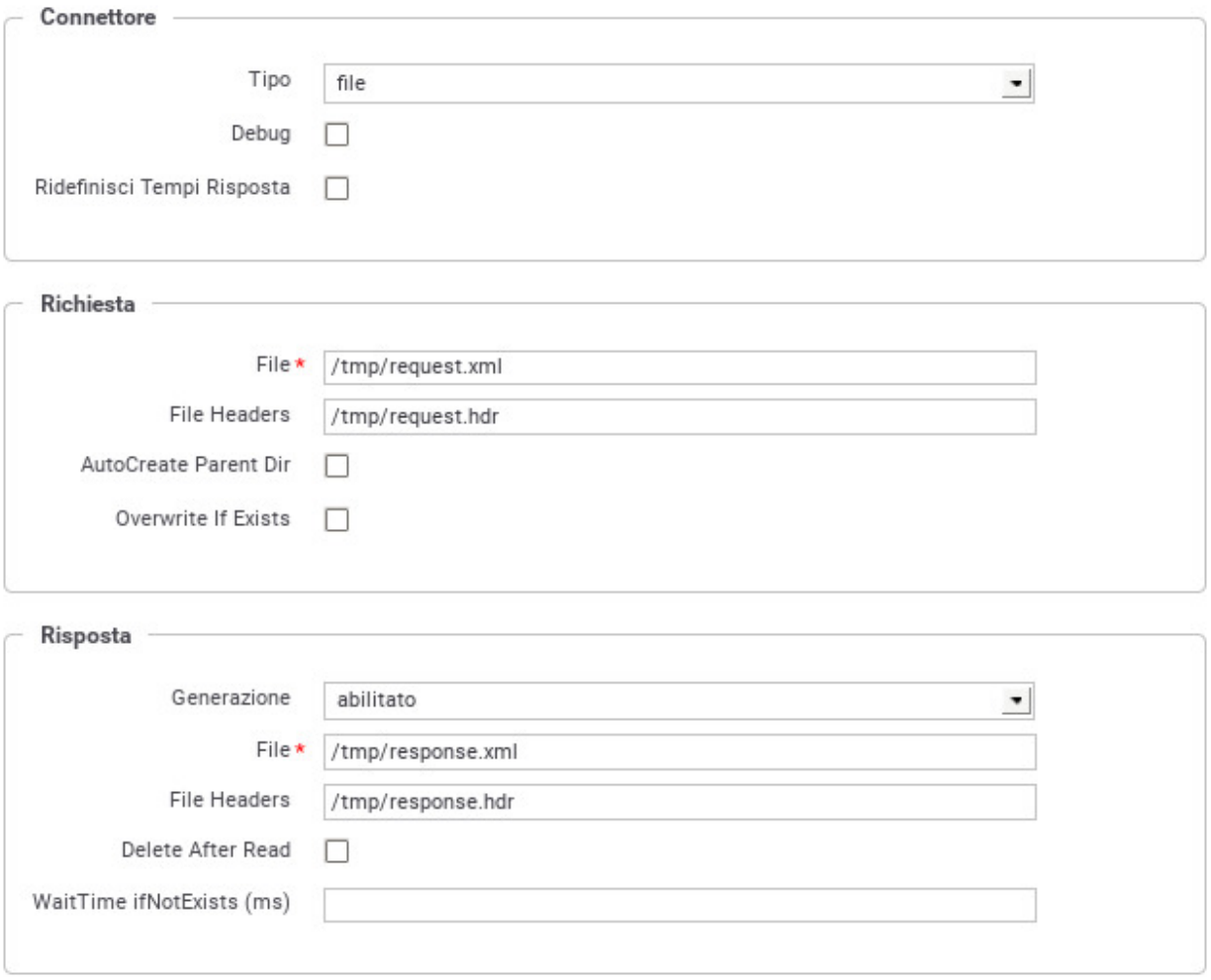


Fig. 10.17: Dati di configurazione di un connettore File

Facendo riferimento alla maschera raffigurata in Fig. 10.17 andiamo a descrivere il significato dei parametri:

- *Richiesta*
  - **File**: indirizzo su file-system (path) dove verrà serializzato il messaggio di richiesta. È possibile fornire delle macro per creare dei path dinamici (per ulteriori dettagli vedi sezione "Informazioni Dinamiche").
  - **File Headers** (opzionale): indirizzo su file-system (path) dove verranno serializzati gli header di trasporto associati alla richiesta. È possibile fornire delle macro per creare dei path dinamici (per ulteriori dettagli vedi sezione "Informazioni Dinamiche").
  - **Overwrite If Exists** (true/false): abilita l'eventuale sovrascrittura del file, se risulta già esistere.
  - **AutoCreate Parent Directory** (true/false): abilita la creazione automatica delle directory padre, se non esistono.
- *Risposta (Opzionale)*
  - **Generazione** (true/false): abilita la generazione di una risposta. Tutte le successive opzioni della sezione "Risposta" sono configurabili solamente se la generazione è abilitata.
  - **File**: indirizzo su file-system (path) dove verrà letto il messaggio di risposta. È possibile fornire delle macro per creare dei path dinamici, come descritto più avanti al punto «Informazioni Dinamiche».
  - **File Headers** (opzionale): indirizzo su file system (path) dove verranno letti gli header di trasporto da associare alla risposta. E" possibile fornire delle macro per creare dei path dinamici, come descritto più avanti al punto «Informazioni Dinamiche».
  - **Delete After Read** (true/false): abilita l'eventuale eliminazione del file una volta utilizzato per la generazione della risposta.
  - **Wait Time If Not Exists (ms)** (opzionale): indica un tempo di attesa (ms) qualora il file per la generazione della risposta non sia immediatamente disponibile.
- *Informazioni Dinamiche*. Per creare dei path dinamici rispetto alla transazione in corso di elaborazione, possono essere utilizzate le seguenti macro:
  - **{date:FORMAT}** indica la data di elaborazione del messaggio. Il formato fornito deve essere conforme a quanto richiesto dalla classe java "java.text.SimpleDateFormat". Ad esempio: {date:yyyyMMdd_HHmmssSSS}.
  - **{transaction:id}** indica l'identificativo della transazione (UUID).
  - **{busta:FIELD}** permette di utilizzare informazioni di protocollo riguardanti la transazione in corso. Il valore "FIELD" fornito deve rappresentare un field valido all'interno della classe di openspcoop "org.openspcoop2.protocol.sdk.Busta". Ad esempio per ottenere il mittente della busta usare {busta:mittente}.
  - **{header:NAME}** permette di utilizzare informazioni, relative alla transazione in corso, inserite negli header http generati da GovWay (maggiori dettagli in sezione *Header di Integrazione*). Il valore "NAME" indica il nome dell'header da utilizzare. Ad esempio per utilizzare il nome del mittente è possibile usare {header:GovWay-Sender}. Un altro esempio valido nello scenario della fatturazione elettronica (sezione *Profilo "Fatturazione Elettronica"*) potrebbe essere quello di utilizzare il nome originale del file fattura utilizzando la sintassi {header:GovWay-SDI-NomeFile}
  - **{query:NAME}** permette di utilizzare informazioni, relative alla transazione in corso, inserite nei query parameter aggiunti all'endpoint da GovWay (maggiori dettagli in sezione *Header di Integrazione*). Il valore "NAME" indica il nome della proprietà da utilizzare. Ad esempio per utilizzare il nome del mittente è possibile usare {query:govway_sender}.
  - **{property:NAME}** permette di utilizzare informazioni, relative alla transazione in corso, specifiche della sezione relativa al profilo utilizzato all'interno della traccia (es. sezione "Informazioni Fatturazione Elettronica"). Il valore "NAME" indica il nome della proprietà da utilizzare. Un esempio valido nello

scenario della fatturazione elettronica (sezione *Profilo "Fatturazione Elettronica"*) potrebbe essere quello di utilizzare l'identificativo sdi utilizzando la sintassi {property:IdentificativoSdI}

10.8 Device PKCS11

Nella sezione pkcs11Install è documentato come configurare GovWay per poter utilizzare token PKCS11.

Una volta registrati, i token saranno selezionabili tra i tipi di keystore disponibili (es. Fig. 10.18) per tutte le funzionalità che richiedono l'utilizzo di una chiave X.509.

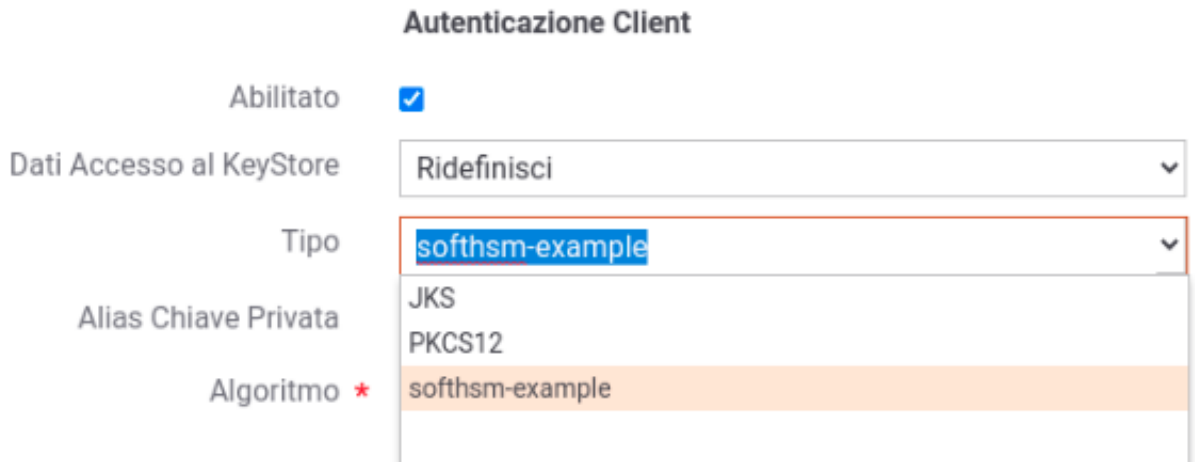


Fig. 10.18: Esempio di configurazione di un token PKCS11 su connettore https

Nota: Le funzionalità che richiedono l'utilizzo della parte pubblica di un certificato X.509 non consentiranno di selezionare i keystore PKCS11 registrati, a meno che durante la registrazione non siano state abilitate le opzioni "usableAsTrustStore" e "usableAsSecretKeyStore". Per maggiori dettagli si rimanda alla sezione pkcs11Install.

10.9 Correlazione tra transazioni differenti

Richiede accesso alla govwayConsole in modalità *avanzata* (sezione *Modalità Avanzata*).

Come descritto anche nella sezione *Configurazione manuale delle interfacce*, durante la configurazione di un API di tipo SOAP o REST è possibile specificare i parametri descritti di seguito rispettivamente in un servizio/azione o in una risorsa.

- *ID Conversazione*. Flag per consentire di specificare nelle richieste un valore che identifica una conversazione.
- *Riferimento ID Richiesta*. Flag per consentire di specificare nelle richieste un identificativo relativo ad un messaggio precedente.

Tali parametri consentono agli applicativi client di fornire tali informazioni tramite gli header di integrazione descritti nella sezione *Scambio di informazioni nella richiesta del client verso il gateway*

Le informazioni fornite saranno associate alla traccia della transazione gestita, e quindi utilizzabili in fase di monitoraggio tramite le modalità di ricerca basate su identificativi descritte nella Guida alla Console di Monitoraggio.

10.10 Opzioni Avanzate per Erogazioni/Fruizioni

A partire dall'erogazione o fruizione di una API, accedendo alla sezione Configurazione, descritta nella sezione *Configurazione dell'API*, in modalità avanzata compare una sezione precedentemente non documentata denominata *Opzioni Avanzate*.

All'interno di tale sezione è possibile configurare (Fig. 10.19):

- *Integrazione - Metadati*: per default non impostato, consente di attivare gli header di integrazione desiderati utilizzando le keyword, separate da virgola, descritta nella sezione *Altri header di Integrazione*.
- *SOAP With Attachments - Gestione Body*: presente solamente per API di tipo SOAP consente tramite la voce "allega" di spostare il contenuto presente nel body in un attachment o di eliminare il body dalla richiesta prima di inoltrare il messaggio.

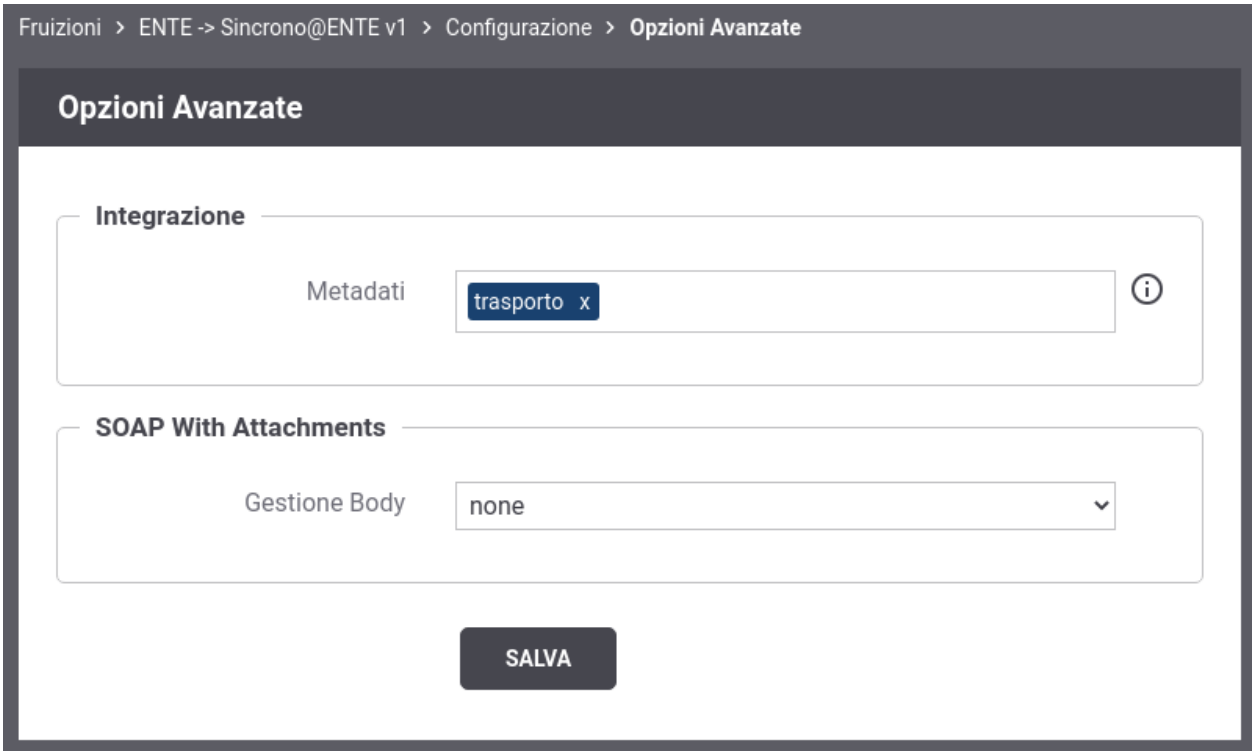


Fig. 10.19: Opzioni Avanzate di una API

10.11 Tracciatura su File

Le informazioni inerenti le comunicazioni gestite dal gateway vengono registrate su una base dati di tracciamento (*Tracciamento*) e sono consultabili tramite una console di monitoraggio (mon_intro).

È possibile estendere il normale tracciamento su database, attivando il tracciamento su file.

La nuova funzionalità consente il tracciamento su file di tutte le informazioni relative alle comunicazioni gestite da GovWay. Il successivo processamento del file da strumenti esterni (es. FileBeat) permette così una facile integrazione con sistemi di tracciamento esterni (es. Logstash, Kafka, . . .).

La funzionalità consente una completa personalizzazione delle informazioni da riportare su file di log, permettendo anche di definirne il formato e l'ordine in cui vengono salvate. È inoltre possibile suddividere le informazioni in più file di log in modo da facilitare l'invio di informazioni selezionate a destinazioni diverse.

Le informazioni possono essere riversate in uno o più topic, dove ad ogni topic corrisponde tipicamente un file di log. Di seguito un esempio di file prodotto:

```
"req"|"b6cdd758-342c-4599-ae95-33a781730b3f"|"govway"|"2020-06-26␣
↪12:46:50:629"|"+0200"|"192.168.1.2"|"HTTP/1.1"|"POST"|"https://server:8446/
↪example"|"application/soap+xml; charset=UTF-8; action=\"test/\""|"10490"|
↪"200"
"req"|"2a9dc253-9dd5-458b-8689-edee7c9ba139"|"govway"|"2020-06-26␣
↪12:47:50:561"|"+0200"|"192.168.1.2"|"HTTP/1.1"|"POST"|"https://server:8446/
↪app2"|"application/soap+xml; charset=UTF-8; action=\"test/\""|"1090"|"503"
"req"|"eeedb92b-66b5-451e-8266-ade2cf1f34ce"|"govway"|"2020-06-26␣
↪12:47:53:291"|"+0200"|"192.168.1.19"|"HTTP/1.1"|"POST"|"https://
↪server:8446/example"|"application/soap+xml; charset=UTF-8; action=\"test/\"
↪"|"11230"|"200"
"req"|"b4355a45-71cc-4293-b3b7-a4622af8ea84"|"govway"|"2020-06-26␣
↪12:48:00:102"|"+0200"|"192.168.1.22"|"HTTP/1.1"|"POST"|"https://
↪server:8446/example"|"application/soap+xml; charset=UTF-8; action=\"test/\"
↪"|"17999"|"200"
```

Per attivare la funzionalità a livello globale abilitare la proprietà "org.openspcoop2.pdd.transazioni.fileTrace.enabled" nel file di configurazione locale "/etc/govway/govway_local.properties" (assumendo sia /etc/govway la directory di configurazione indicata in fase di installazione). Una volta attivata, si deve indicare, attraverso la proprietà "org.openspcoop2.pdd.transazioni.fileTrace.config", dove reperire il file di configurazione che definisce il formato dei log prodotti dal gateway (formato descritto in *Configurazione dei Topic*). Di seguito un estratto della configurazione.

```
# ================================================
# FileTrace
# Indicazione se la funzionalità è abilitata o meno
org.openspcoop2.pdd.transazioni.fileTrace.enabled=true
#
...
#
# File di Configurazione
# Il file può essere indicato con un path assoluto o relativo rispetto alla␣
↪directory di configurazione
org.openspcoop2.pdd.transazioni.fileTrace.config=govway.fileTrace.properties
# ================================================
```

Per attivare la funzionalità e/o modificare la configurazione utilizzata di FileTrace su una singola API si possono registrare le seguenti *Proprietà* sull'erogazione o sulla fruizione:

- *fileTrace.enabled* : consente di attivare o disattivare la funzionalità (i valori associabili alla proprietà sono "true" o "false");
- *fileTrace.config* : consente di indicare il path su file system dove risiede la configurazione della tracciatura su file; il file indicato può essere un path assoluto o relativo rispetto alla directory di configurazione (per il formato fare riferimento alla sezione *Configurazione dei Topic*);
- *fileTrace.dumpBinario.enabled*: consente di attivare o disattivare la registrazione dei messaggi scambiato con il client: richiesta ingresso e risposta uscita (i valori associabili alla proprietà sono "true" o "false");
- *fileTrace.dumpBinario.connettore.enabled*: consente di attivare o disattivare la registrazione dei messaggi scambiato con l'implementazione di backend dell'API: richiesta uscita e risposta ingresso (i valori associabili alla proprietà sono "true" o "false").

Nota: Solamente se la registrazione (*fileTrace.dumpBinario.enabled* e/o *fileTrace.dumpBinario.connettore.enabled*) è abilitata sarà possibile accedere ai contenuti dei messaggi come descritto nella sezione *Informazioni Tracciabili*.

Nella sezione *Configurazione dei Topic* viene descritto il formato del file di configurazione, mentre nella sezione *Informazioni Tracciabili* sono riportate tutte le informazioni disponibili.

10.11.1 Configurazione dei Topic

Le informazioni inerenti le comunicazioni gestite dal gateway possono essere riversate in uno o più file di log attraverso la definizione di topic.

La configurazione permette di indicare uno o più topic da generare quando il gateway gestisce erogazioni o fruizioni di API. Nell'esempio seguente vengono registrati due topic sulle erogazioni, dove si vuole salvere le informazioni suddividendole tra richiesta e la risposta. Per quanto concerne la registrazione delle fruizioni si attiva invece solamente un unico topic.

```
# Topic
topic.erogazioni=inputRequest,inputResponse
topic.fruizioni=output
```

Per default tutte le comunicazioni gestite dal gateway vengono veicolati nei topic registrati. È possibile escludere il riversamento nel topic di determinate comunicazioni tramite le seguenti proprietà:

- *log.topic.*<*erogazioni/fruizioni*>.<*nomeTopic*>.*requestSended* : se abilita, sul topic indicato verranno riversate solamente informazioni relative a comunicazioni per le quali il gateway è riuscito a spedire la richiesta verso il backend configurato;
- *log.topic.*<*erogazioni/fruizioni*>.<*nomeTopic*>.*[in/out][Request/Response]ContentDefined* : se abilitata, verranno riversate informazioni sul topic solo se la richiesta o risposta indicata, in ingresso o uscita dal gateway, possiede un payload http.

Nell'esempio seguente il topic relativo alle fruizioni viene alimentato solamente se il gateway è riuscito a contattare il backend e la richiesta possedeva un payload http (vengono escluse ad esempio le HTTP GET). Sui topic delle erogazioni viene invece attivato solamente il controllo sul payload http per il topic "inputRequest".

```
# Erogazioni (Filtro per Payload HTTP)
topic.erogazioni.inputRequest.inRequestContentDefined=true

# Fruizioni (Filtro per RequestSended + Payload HTTP)
topic.fruizioni.output.requestSended=true
topic.fruizioni.output.outRequestContentDefined=true
```

La generazione dei file di log è gestita dalle seguenti proprietà:

- *log.config.file* : file contenente la configurazione *log4j2* nella quale devono essere definite le Category da associare ad ogni topic;
- *log.severity* (default: info): indica il livello di severità (trace/debug/info/warn/error) utilizzato durante il logging;
- *log.topic.*<*erogazioni/fruizioni*>.<*nomeTopic*>=<*categoryLog4j2*> : assegna la category al topic indicato per le erogazioni o fruizioni.

Nell'esempio seguente viene fornito un esempio di associazione di Category ad ogni topic e un estratto di configurazione log4j2 nella quale viene creato un file per ogni category.

```
# Log4j2 Configuration File
log.config.file=govway.fileTrace.log4j2.properties

# trace/debug/info/warn/error
log.severity=info

# Category per ogni topic delle erogazioni
category.topic.erogazioni.inputRequest=fileTrace.inputRequest
category.topic.erogazioni.inputResponse=fileTrace.inputResponse
# Category per ogni topic delle fruizioni
# sintassi: log.topic.fruizioni.<nomeTopic>=<categoryLog4j2>
category.topic.fruizioni.output=fileTrace.output
```

Estratto della configurazione Log4J2 dove per ogni category viene attivata una rotazione giornaliera:

```
name = fileTracePropertiesConfig

# ** inputRequest **
# Category
logger.fileTrace_inputRequest.name = fileTrace.inputRequest
logger.fileTrace_inputRequest.level = DEBUG
logger.fileTrace_inputRequest.additivity = false
logger.fileTrace_inputRequest.appenderRef.rolling.ref = fileTrace.
↪inputRequest.rollingFile
# FileAppender
appender.fileTrace_inputRequest.type = RollingFile
appender.fileTrace_inputRequest.name = fileTrace.inputRequest.rollingFile
appender.fileTrace_inputRequest.fileName = /var/govway/log/fileTrace/
↪inputRequest.log
appender.fileTrace_inputRequest.filePattern = /var/govway/log/fileTrace/$$
↪{date:yyyy-MM}/inputRequest-%d{MM-dd-yyyy}.log.gz
appender.fileTrace_inputRequest.layout.type = PatternLayout
appender.fileTrace_inputRequest.layout.pattern = %m%n
appender.fileTrace_inputRequest.policies.type = Policies
appender.fileTrace_inputRequest.policies.time.type =␣
↪TimeBasedTriggeringPolicy
appender.fileTrace_inputRequest.strategy.type = DefaultRolloverStrategy

# ** inputResponse**
# Category
logger.fileTrace_inputResponse.name = fileTrace.inputResponse
logger.fileTrace_inputResponse.level = DEBUG
logger.fileTrace_inputResponse.additivity = false
logger.fileTrace_inputResponse.appenderRef.rolling.ref = fileTrace.
↪inputResponse.rollingFile
# FileAppender
appender.fileTrace_inputResponse.type = RollingFile
appender.fileTrace_inputResponse.name = fileTrace.inputResponse.rollingFile
appender.fileTrace_inputResponse.fileName = /var/govway/log/fileTrace/
↪inputResponse.log
appender.fileTrace_inputResponse.filePattern = /var/govway/log/fileTrace/$$
↪{date:yyyy-MM}/inputResponse-%d{MM-dd-yyyy}.log.gz
appender.fileTrace_inputResponse.layout.type = PatternLayout
appender.fileTrace_inputResponse.layout.pattern = %m%n
appender.fileTrace_inputResponse.policies.type = Policies
appender.fileTrace_inputResponse.policies.time.type =␣
↪TimeBasedTriggeringPolicy
```

(continues on next page)

(continua dalla pagina precedente)

```
appender.fileTrace_inputResponse.strategy.type = DefaultRolloverStrategy

# ** output **
# Category
logger.fileTrace_output.name = fileTrace.output
logger.fileTrace_output.level = DEBUG
logger.fileTrace_output.additivity = false
logger.fileTrace_output.appenderRef.rolling.ref = fileTrace.output.
↪rollingFile
# FileAppender
appender.fileTrace_output.type = RollingFile
appender.fileTrace_output.name = fileTrace.output.rollingFile
appender.fileTrace_output.fileName = /var/govway/log/fileTrace/output.log
appender.fileTrace_output.filePattern = /var/govway/log/fileTrace/$$
↪{date:yyyy-MM}/output-%d{MM-dd-yyyy}.log.gz
appender.fileTrace_output.layout.type = PatternLayout
appender.fileTrace_output.layout.pattern = %m%n
appender.fileTrace_output.policies.type = Policies
appender.fileTrace_output.policies.time.type = TimeBasedTriggeringPolicy
appender.fileTrace_output.strategy.type = DefaultRolloverStrategy
```

Per ogni topic non rimane che definire le informazioni che si desidera tracciare attraverso la proprietà "*format.topic.<erogazioni/fruizioni>.<nomeTopic>*". Le informazioni possono essere definite attraverso costanti o tramite quanto indicato nella sezione *Informazioni Tracciabili*.

Di seguito un esempio:

```
format.topic.erogazioni.inputRequest="req"|"${log:transactionId}"|"govway"|"$
↪{log:inRequestDateZ(yyyy-MM-dd HH:mm:ss:SSS,UTC)}"|"${log:inRequestDate(Z)}
↪"|"${log:forwardedIP}"|"HTTP/1.1"|"${log:httpMethod}"
format.topic.erogazioni.inputResponse="res"|"${log:transactionId}"|"govway"|"
↪${log:inRequestDateZ(yyyy-MM-dd HH:mm:ss:SSS,UTC)}"|"$
↪{log:inRequestDate(Z)}"|"${log:forwardedIP}"|"HTTP/1.1"|"${log:httpMethod}
↪"|"${log:outHttpStatus}"
format.topic.fruizioni.output="output"|"${log:transactionId}"|"govway"|"$
↪{log:inRequestDateZ(yyyy-MM-dd HH:mm:ss:SSS,UTC)}"|"${log:inRequestDate(Z)}
↪"|"${log:forwardedIP}"|"HTTP/1.1"|"${log:httpMethod}"|"${log:inHttpStatus}"
```

Le informazioni prodotte ad esempio per il topic inputRequest saranno le seguenti:

```
"req"|"b6cdd758-342c-4599-ae95-33a781730b3f"|"govway"|"2020-06-26␣
↪12:46:50:629"|"+0200"|"192.168.1.2"|"HTTP/1.1"|"POST"
"req"|"2a9dc253-9dd5-458b-8689-edee7c9ba139"|"govway"|"2020-06-26␣
↪12:47:50:561"|"+0200"|"192.168.1.2"|"HTTP/1.1"|"POST"
"req"|"eeedb92b-66b5-451e-8266-ade2cf1f34ce"|"govway"|"2020-06-26␣
↪12:47:53:291"|"+0200"|"192.168.1.19"|"HTTP/1.1"|"POST"
"req"|"b4355a45-71cc-4293-b3b7-a4622af8ea84"|"govway"|"2020-06-26␣
↪12:48:00:102"|"+0200"|"192.168.1.22"|"HTTP/1.1"|"POST"
```

Nell'esempio appena riportato si può notare come i 3 topic utilizzano una parte comune. È possibile ottimizzare le informazioni configurate attraverso la definizione di proprietà "*format.property.<posizione>.<nomeProprietà>=<valoreProprietà>*". Le proprietà verranno risolte in ordine lessicografico rispetto alla posizione indicata, in modo da garantire la corretta risoluzione se si hanno proprietà che sono definite tramite altre proprietà.

Di seguito il precedente esempio ridefinto tramite proprietà:

```
# properties
format.property.001.commons.govway-id=govway
format.property.001.commons.id="${log:transactionId}"|"$
↪{log:property(commons.govway-id)}"
format.property.002.commons.data="${log:inRequestDateZ(yyyy-MM-dd␣
↪HH:mm:ss:SSS,UTC)}"|"${log:inRequestDate(Z)}"
format.property.003.commons.remoteIP-protocol-method="${log:forwardedIP}"|
↪"HTTP/1.1"|"${log:httpMethod}"
format.property.004.commons=${log:property(commons.id)}|$
↪{log:property(commons.data)}|${log:property(commons.remoteIP-protocol-
↪method)}

# topic
format.topic.erogazioni.inputRequest="req"|${log:property(commons)}
format.topic.erogazioni.inputResponse="res"|${log:property(commons)}|"$
↪{log:outHttpStatus}"
format.topic.fruizioni.output="output"|${log:property(commons)}|"$
↪{log:inHttpStatus}"
```

È infine possibile definire l'escape di caratteri che possono essere presenti nelle informazioni da tracciare tramite la proprietà "*format.escape.<char>=<charEscaped>*".

Di seguito un esempio di configurazione che effettua l'escape del carattere "" sostituendolo con "\»":

```
format.escape."=\\"
```

Nota: In caso di configurazione globale (attivata da file govway_local.properties come indicato in *Tracciatura su File*), anche se la configurazione viene modificata non sarà utilizzata dal Gateway fino ad un suo riavvio. È possibile forzare la rilettura immediata accendendo alla voce "Strumenti - Runtime" della console di gestione e selezionando "Aggiorna la configurazione" nella sezione «Informazioni Tracciamento - File Trace" (Fig. 10.20)».

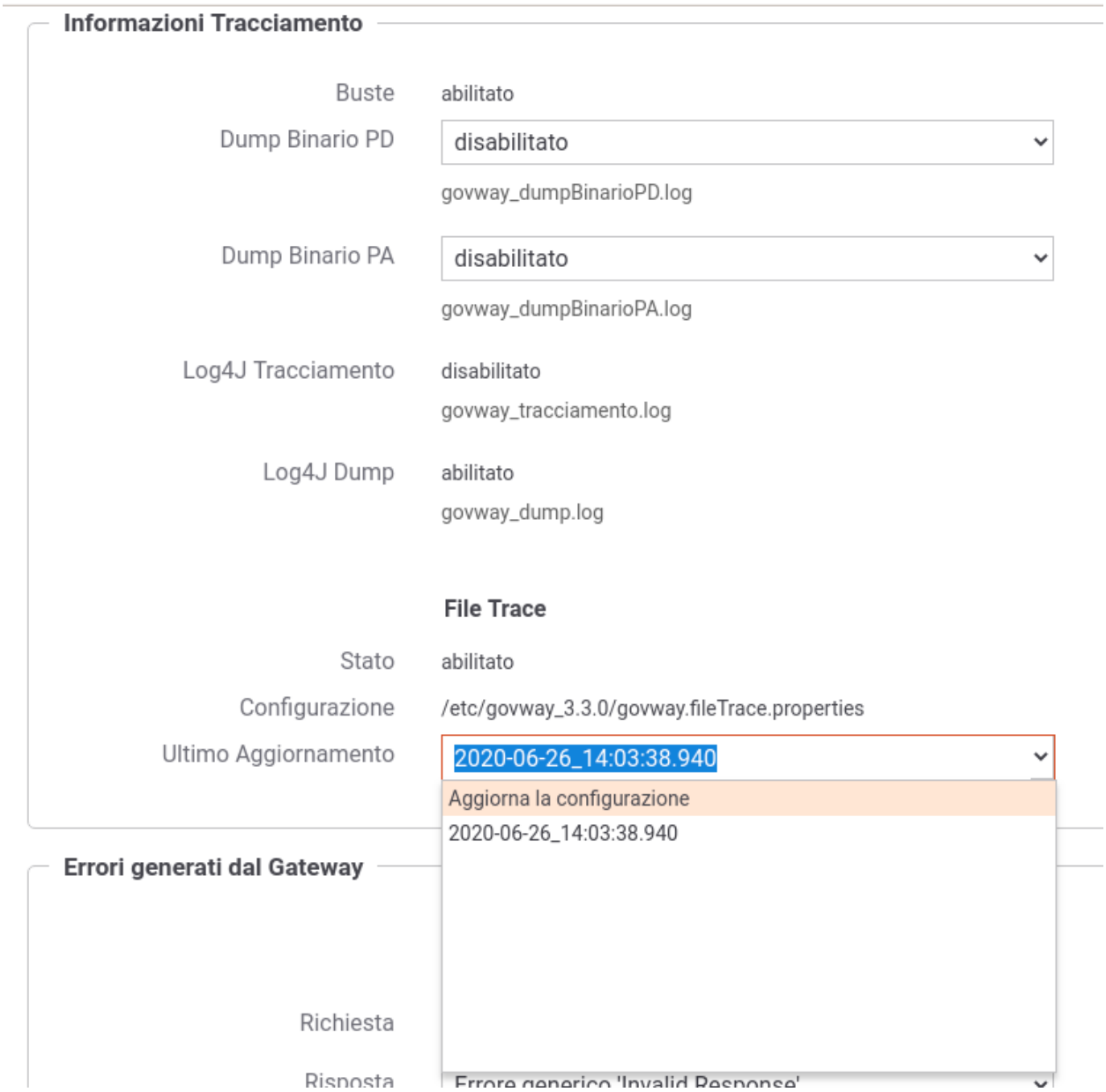


Fig. 10.20: Aggiornamento della Configurazione di File Trace

Anche in caso di configurazione locale (attivata tramite le *Proprietà* come indicato in *Tracciatura su File*) la configurazione modificata non sarà utilizzata dal Gateway fino ad un suo riavvio. È possibile forzare la rilettura immediata accendendo alla voce "Strumenti - Runtime" della console di gestione e cliccando sulla voce "Svuota le Cache".

10.11.2 Informazioni Tracciabili

Le informazioni inerenti le comunicazioni gestite dal gateway, che possono essere riversate nei file di log associati ai topic, sono indicabili all'interno del formato di un topic tramite una delle seguenti sintassi:

- *${log:<id>}* : viene registrata la risorsa con l'identificativo indicato.
- *${log:<id>(defaultValue)}* : viene registrata la risorsa con l'identificativo indicato; se la risorsa non è valorizzata, viene registrato il valore di default fornito come parametro
- *${log:<id>(parameters ...)}* : viene registrata la risorsa con l'identificativo indicato, il cui valore può essere personalizzato rispetto ad alcuni parametri.

Le informazioni possono essere registrate codificate in base64 utilizzando il prefisso "logBase64" invece di "log":

- *${logBase64:<id>}*
- *${logBase64:<id>(defaultValue)}*
- *${logBase64:<id>(parameters ...)}*

L'esempio seguente definisce un topic che utilizza i formati precedentemente indicati. Viene registrato l'identificativo di transazione (informazione acceduta puntualmente), la data di accesso all'API (informazione formattata rispetto ai parametri "yyyy-MM-dd HH:mm:ss:SSS" e "UTC"), il contenuto della richiesta codificato in base64 e l'identificativo di correlazione applicativa se presente o la costante "ExampleDefaultValue" altrimenti.

```
format.topic.erogazioni.example=${log:transactionId}|$
↪{log:inRequestDateZ(yyyy-MM-dd HH:mm:ss:SSS,UTC):ss:SSS,UTC)}"|$
↪{logBase64:inRequestContent}|${log:applicationId(ExampleDefaultValue)}|
```

Le informazioni prodotte saranno le seguenti:

```
"b6cdd758-342c-4599-ae95-33a781730b3f"|"2020-06-26 12:46:50:629
↪"|eyJtaXR0ZW50ZSI6IkF2dm9jYXR1cmEgR2VuZXJhbGUgZGVsbG8gU3RhdG8iLCJkZXN0a...
↪|ExampleDefaultValue
"2a9dc253-9dd5-458b-8689-edee7c9ba139"|"2020-06-26 12:47:50:561
↪"|eyJtaXR0ZW50ZSI6IkF2dm9jYXR1cmEgR2VuZXJhbGUgZGVsbG8gU3RhdG8iLCJkZXN0a...
↪|ExampleDefaultValue
"eeedb92b-66b5-451e-8266-ade2cf1f34ce"|"2020-06-26 12:47:53:291
↪"|eyJtaXR0ZW50ZSI6IkF2dm9jYXR1cmEgR2VuZXJhbGUgZGVsbG8gU3RhdG8iLCJkZXN0a...
↪|ApplicationXXX3
"b4355a45-71cc-4293-b3b7-a4622af8ea84"|"2020-06-26 12:48:00:102
↪"|eyJtaXR0ZW50ZSI6IkF2dm9jYXR1cmEgR2VuZXJhbGUgZGVsbG8gU3RhdG8iLCJkZXN0a...
↪|ExampleDefaultValue
```

Di seguito vengono indicati tutti gli identificativi delle informazioni disponibili con i possibili parametri.

Nota: Gli identificativi per cui non vengono specificati parametri sono sempre disponibili nella modalità con o senza la definizione del valore di default.

Identificativi

- transactionId: identificativo della transazione;
- requestId: identificativo del messaggio di richiesta;
- responseId: identificativo del messaggio di risposta;
- correlationId: identificativo che correla molteplici transazioni;
- asyncId: identificativo utilizzato in profili asincroni;

- requestApplicationId: identificativo di correlazione applicativa della richiesta;
- responseApplicationId: identificativo di correlazione applicativa della risposta;
- applicationId: requestApplicationId + responseApplicationId;
- clusterId: identificativo del nodo in una installazione in cluster del gateway.

Esito

- inHttpStatus: codice http di risposta ritornato dal backend contattato dal gateway;
- inHttpReason: http reason associato al codice di risposta ritornato dal backend;
- outHttpStatus: codice http di risposta ritornato al client dal gateway;
- outHttpReason: http reason associato al codice di risposta ritornato al client;
- resultClass: classe a cui appartiene l'esito della transazione tra OK, KO e FAULT;
- resultClassOk: indicazione se l'esito della transazione appartiene alla classe OK (true/false);
- resultClassKo: indicazione se l'esito della transazione appartiene alla classe KO (true/false);
- resultClassFault: indicazione se l'esito della transazione appartiene alla classe FAULT (true/false);
- result: esito della transazione (codifica GovWay);
- errorDetail: dettaglio dell'errore avvenuto durante la gestione della transazione;
- transactionType: tipo della transazione (Standard, Sistema . . .).

Diagnostici

Di seguito vengono indicati gli identificativi che consentono di accedere ai diagnostici emessi da GovWay durante la gestione della richiesta:

- diagnostics: elenco completo dei messaggi diagnostici emessi;
- errorDiagnostics: elenco dei messaggi diagnostici di sola severità errore.

Ogni diagnostico viene fornito nella forma seguente e separato dagli altri tramite un ritorno a capo (configurazione di default):

<livelloSeverità> <dataEmissione> <codiceDiagnostico> <messaggio>

ad esempio:

```
infoIntegration 2020-09-10T14:15:51.605Z 004074 Autenticazione [basic] in␣
↪corso ( BasicUsername 'prova' ) ...
infoIntegration 2020-09-10T14:15:51.606Z 004075 Autenticazione [basic]␣
↪effettuata con successo (in cache)
```

L'elenco dei diagnostici sono accessibili anche con i seguenti parametri:

- (separator): consente di indicare un separatore dei diagnostici differente da quello di default (ritorno a capo)
- (separator, format): oltre al separatore, consente di indicare il formato della data (es. yyyy-MM-dd HH:mm:ss:SSS.Z);
- (separator, format, timeZone): oltre al separatore e al formato della data (es. yyyy-MM-dd HH:mm:ss:SSS) consente di indicare il time zone (es. UTC).

Date

- acceptedRequestDate: data in cui la richiesta è pervenuta sul gateway;
- inRequestDate: data in cui la richiesta è stata completamente ricevuta sul gateway;

- outRequestDate: data in cui la richiesta viene inoltrata dal gateway al backend;
- acceptedResponseDate: data in cui la risposta è pervenuta sul gateway;
- inResponseDate: data in cui la risposta è stata completamente ricevuta sul gateway;
- outResponseDate: data in cui la risposta viene ritornata al client.

Tutte le date indicate sono accessibili anche con i seguenti parametri:

- (format): formato della data (es. yyyy-MM-dd HH:mm:ss:SSS.Z);
- (format, timeZone): formato della data (es. yyyy-MM-dd HH:mm:ss:SSS) + time zone (es. UTC).

Elapsed Time

- elapsedTime: tempo di risposta complessivo trascorso tra l'ingresso della richiesta nel gateway e la risposta ritornata al client;
- apiElapsedTime: tempo di risposta del backend;
- gatewayLatency: latenza introdotta dal gateway rispetto al tempo di risposta del backend.

Tutte le informazioni sono ritornate in millisecondi. È possibile ottenere le medesime informazioni in un altro formato di tempo utilizzando i seguenti suffissi:

- <elapsedTime>S: tempo in secondi;
- <elapsedTime>Ms: tempo in millisecondi (è il default);
- <elapsedTime>uS: tempo in microsecondi;
- <elapsedTime>nS: tempo in nanosecondi.

Dominio

- domain: identificativo del dominio interno che ha gestito l'erogazione o la fruizione;
- organizationId: identificativo del soggetto, di dominio interno, che ha gestito l'erogazione o la fruizione (identificativo nel formato previsto dal profilo di interoperabilità);
- organization: nome del soggetto, di dominio interno, che ha gestito l'erogazione o la fruizione;
- organizationType: tipo del soggetto, di dominio interno, che ha gestito l'erogazione o la fruizione;
- role: indica se la transazione rappresenta una "erogazione" o "fruizione";
- contextPropertiesKeys: nomi delle proprietà definite nel contesto;
- contextProperties: proprietà (nome=valore) definite nel contesto separate da uno spazio;
- contextProperties(propertySeparator, valueSeparator): simile alla precedente opzione, consente di indicare i separatori utilizzati;
- contextProperty(nomeProprietà): valore della proprietà indicata come parametro.

API

- apiProtocol: indica se l'API è di tipo "rest" o "soap";
- apiId: identificativo dell'API, secondo il formato previsto dal profilo di interoperabilità;
- api: nome dell'API;
- apiVersion: versione dell'API;
- apiType: tipo dell'API;
- apiInterface: identificativo dell'interfaccia implementata dall'erogazione o dalla fruizione (contiene nome, versione e soggetto referente);

- apiInterfaceId: identificativo dell'interfaccia implementata dall'erogazione o dalla fruizione secondo il formato previsto dal profilo di interoperabilità;
- apiPropertiesKeys: nomi delle proprietà definite sull'erogazione o sulla fruizione;
- apiProperties: proprietà (nome=valore) definite sull'erogazione o sulla fruizione separate da uno spazio;
- apiProperties(propertySeparator, valueSeparator): simile alla precedente opzione, consente di indicare i separatori utilizzati;
- apiProperty(nomeProprietà): valore della proprietà indicata come parametro;
- action: identificativo della risorsa (API Rest) o dell'azione (API Soap);
- httpMethod: metodo http invocato;
- outURL: url utilizzata dal gateway per invocare il backend (se presenti, contiene anche i parametri della url);
- inURL: url utilizzata dal client per invocare il gateway (se presenti, contiene anche i parametri della url);
- inFunction: indica il tipo di canale (in, out, out/xml2soap) utilizzato dal client per invocare il gateway;
- collaborationProfileCode: indica il profilo di collaborazione associato all'azione di una API Soap (Oneway/Sincrono/AsincronoSimmetrico/AsincronoAsimmetrico);
- collaborationProfile: indica il profilo di collaborazione associato all'azione di una API Soap con la terminologia del profilo di interoperabilità dell'API;
- profile: profilo di interoperabilità in cui è stata registrata l'API;
- profileLabel: nome descrittivo del profilo di interoperabilità in cui è stata registrata l'API;
- interface: identificativo dell'erogazione o della fruizione;
- outConnectorName: nome del connettore multiplo selezionato per la consegna.

Soggetti

- providerId: identificativo del soggetto erogatore, secondo il formato previsto dal profilo di interoperabilità;
- provider: nome del soggetto erogatore;
- providerType: tipo del soggetto erogatore;
- providerDomain: identificativo del dominio erogatore;
- providerURI: uri associata al soggetto erogatore;
- providerPropertiesKeys: nomi delle proprietà definite sul soggetto fruitore;
- providerProperties: proprietà (nome=valore) definite sul soggetto fruitore separate da uno spazio;
- providerProperties(propertySeparator, valueSeparator): simile alla precedente opzione, consente di indicare i separatori utilizzati;
- providerProperty(nomeProprietà): valore della proprietà indicata come parametro;
- senderId: identificativo del soggetto fruitore, secondo il formato previsto dal profilo di interoperabilità;
- sender: nome del soggetto fruitore;
- senderType: tipo del soggetto fruitore;
- senderDomain: identificativo del dominio fruitore;
- senderURI: uri associata al soggetto fruitore;
- senderPropertiesKeys: nomi delle proprietà definite sul soggetto fruitore;
- senderProperties: proprietà (nome=valore) definite sul soggetto fruitore separate da uno spazio;

- senderProperties(propertySeparator, valueSeparator): simile alla precedente opzione, consente di indicare i separatori utilizzati;
- senderProperty(nomeProprietà): valore della proprietà indicata come parametro.

Mittente

- application: identificativo dell'applicativo richiedente;
- applicationPropertiesKeys: nomi delle proprietà definite sull'applicativo richiedente;
- applicationProperties: proprietà (nome=valore) definite sull'applicativo separate da uno spazio;
- applicationProperties(propertySeparator, valueSeparator): simile alla precedente opzione, consente di indicare i separatori utilizzati;
- applicationProperty(nomeProprietà): valore della proprietà indicata come parametro;
- credentials: credenziali presenti nella richiesta;
- principal: identificato con cui l'applicativo è stato autenticato;
- principalAuthType: tipo di autenticazione (basic/ssl/principal) con cui l'applicativo è stato autenticato;
- token: token OAuth2 presente nella richiesta;
- tokenIssuer: issuer presente nel token;
- tokenSubject: subject presente nel token;
- tokenClientId: clientId presente nel token;
- tokenUsername: username presente nel token;
- tokenMail: eMail presente nel token;
- attribute(nomeAttributo): valore dell'attributo indicato come parametro (informazione disponibile solamente se nell'erogazione/fruizione è stata configurata una sola A.A.);
- attributeByAA(nomeAttributeAuthority,nomeAttributo): valore dell'attributo recuperato tramite l'AttributeAuthority indicata come parametro (informazione disponibile solamente se nell'erogazione/fruizione è stata configurata più di una A.A.);
- clientIP: indirizzo IP del client;
- forwardedIP: indirizzo IP presente nella richiesta in uno degli header http appartenente alla classe «Forwarded-For» o «Client-IP»;
- requester: rappresenta il richiedente della richiesta e assumerà la prima informazione valorizzata, trovata nella richiesta, nel seguente ordine:
  - tokenUsername: username presente nel token
  - tokenSubject[@tokenIssuer]: subject presente nel token; viene aggiunto anche un suffisso @tokenIssuer se è presente anche un issuer nel token
  - application: identificativo dell'applicativo richiedente
  - principal: identificato (credenziali) con cui l'applicativo è stato autenticato; se il tipo di autenticazione risulta essere "ssl" viene ritornato il valore dell'attributo CN
  - tokenClientId: clientId presente nel token
- ipRequester: rappresenta l'indirizzo ip del richiedente e viene valorizzato con il forwardedIP se presente, o altrimenti con il clientIP.

Messaggi

- duplicateRequest: numero di volte in cui una richiesta con stesso "requestId" è stata ricevuta dal gateway;
- duplicateResponse: numero di volte in cui una risposta con stesso "responseId" è stata ricevuta dal gateway;
- getInFault: eventuale SOAP Fault o Problem Detail RFC 7807 ricevuto dal backend;
- getOutFault: eventuale SOAP Fault o Problem Detail RFC 7807 ritornato al client.

È inoltre possibile accedere alle seguenti informazioni riguardanti i singoli messaggi in ingresso o uscita dal gateway:

- <messageType>ContentType: valore dell'header http "Content-Type";
- <messageType>Content: payload http;
- <messageType>Size: dimensione del payload http;
- <messageType>Header(name): valore dell'header http indicato come parametro;
- <messageType>Header(name, multiValueSeparator): elenco di valori, separati con il carattere indicato nel parametro "multiValueSeparator", relativi agli header http che possiedono il nome indicato dal parametro "name";
- <messageType>Headers: elenco degli headers http nel formato <nome>=<valore> separati dal carattere "," ;
- <messageType>Headers(headersSeparator, nameValueSeparator, prefix, suffix): i parametri permettono di personalizzare il formato degli headers http.

I tipi di messaggi disponibili sono:

- inRequest: richiesta ricevuta sul gateway;
- outRequest: richiesta inoltrata al backend;
- inResponse: risposta ricevuta dal backend;
- outResponse: risposta ritornata a client.

Nota: Le informazioni sui 4 tipi di messaggio saranno disponibili solamente se è stata abilitata la funzionalità di dump per ciascun tipo nel file di configurazione locale "/etc/govway/govway_local.properties" (assumendo sia /etc/govway la directory di configurazione indicata in fase di installazione) o tramite le *Proprietà* come indicato in *Tracciatura su File*. Di seguito un estratto della configurazione globale che riporta l'abilitazione dei 4 tipi:

```
# ================================================
# FileTrace
...
#
# Indicazione se nella funzionalità è consentito l'accesso ai contenuti
# -- Fruizioni --
# inRequest/outResponse
org.openspcoop2.pdd.transazioni.fileTrace.dumpBinarioPD.enabled=true
# outRequest/inResponse
org.openspcoop2.pdd.transazioni.fileTrace.dumpBinarioPD.connettore.
↪enabled=true
# -- Erogazioni --
# inRequest/outResponse
org.openspcoop2.pdd.transazioni.fileTrace.dumpBinarioPA.enabled=true
# outRequest/inResponse
org.openspcoop2.pdd.transazioni.fileTrace.dumpBinarioPA.connettore.
↪enabled=true
...
```

10.12 Gestione Proxy

I connettori descritti nella sezione *Connettori* rappresentano le entità di configurazione che consentono a GovWay di indirizzare le comunicazioni verso gli attori dei flussi di erogazione/fruizione gestiti. Come già descritto in tale sezione possiamo distinguere due tipologie di comunicazioni:

- *GovWay —> Applicativo Esterno*, nel caso di fruizioni
- *GovWay —> Applicativo Interno*, nel caso di erogazioni

In alcune architetture potrebbe essere presente tra GovWay e l'applicativo da contattare un proxy che media le comunicazioni.

- *Proxy HTTP*, se la comunicazione è mediata da un proxy http l'indirizzo remoto dell'applicativo viene censito su GovWay e la mediazione tramite il proxy sarà trasparente seguendo le indicazioni di configurazione descritte nella sezione *Proxy*.
- *Proxy Applicativo*, in scenari più complessi possono essere presenti reverse proxy che intervengono nella gestione delle connessioni https, utilizzando certificati client e/o trustStore differenti per diversi contesti applicativi. In queste situazioni l'endpoint indicato nella configurazione del connettore su GovWay non è l'indirizzo remoto dell'applicativo ma bensì l'indirizzo del reverse proxy il quale a sua volta si occuperà di inoltrare la richiesta agli indirizzi a lui noti. In questa situazione, è necessario configurare gli endpoint delle API sia su GovWay (indirizzo del reverse proxy), che sul reverse proxy (indirizzo dell'Erogatore finale)

Per semplificare la gestione, in uno scenario architetturale con *Proxy Applicativo*, GovWay può passare l'indirizzo remoto dell'applicativo al proxy tramite un header HTTP o un parametro della url. In questo modo il censimento degli applicativi viene effettuato esclusivamente su GovWay.

Per abilitare e configurare la funzionalità "govway-proxy" si deve agire a livello di proprietà java, configurabili accedendo alla sezione "Configurazione Generale -> Proprietà di Sistema", aggiungendo una proprietà "govway-proxy-enable" con valore "true" (Figura Fig. 10.21).

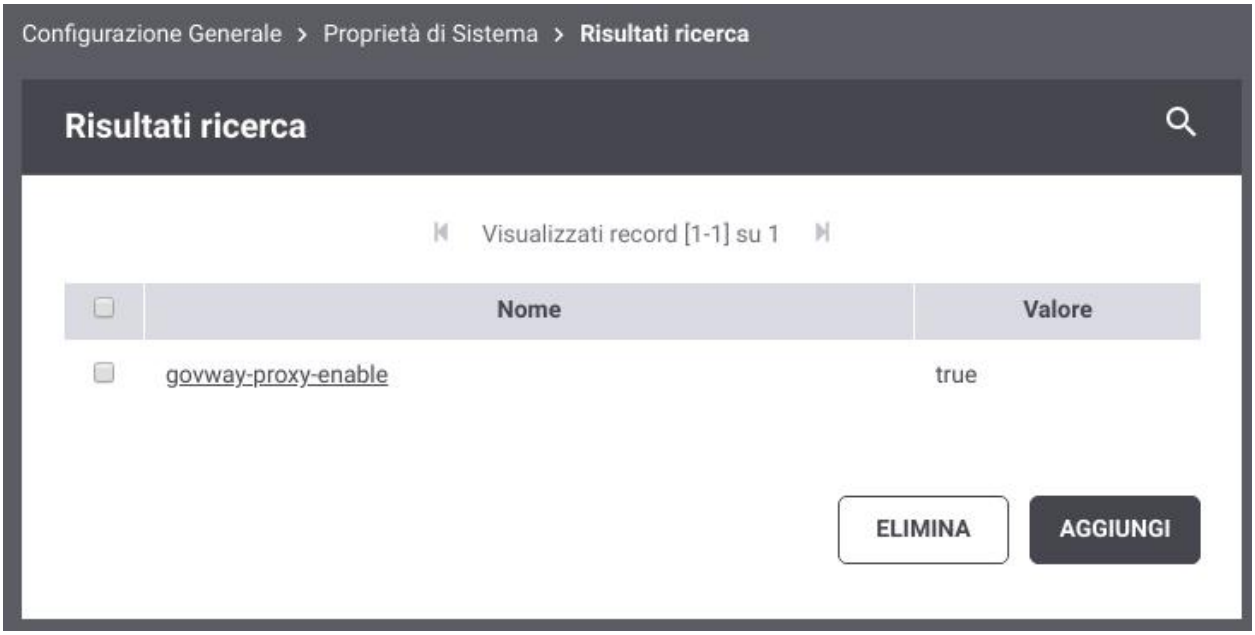


Fig. 10.21: Configurazione delle Proprietà di Sistema

Una volta abilitata la funzionalità la configurazione è attuabile tramite le seguenti proprietà:

- govway-proxy: endpoint a cui verranno inoltrate le richieste. L'endpoint può contenere parti dinamiche che verranno risolte dal Gateway (per ulteriori dettagli fare riferimento alla sezione *Valori dinamici*);
- govway-proxy-header: se configurato verrà utilizzato un header http, con il nome indicato, per inoltrare al proxy l'indirizzo remoto;
- govway-proxy-header-base64: nel caso sia stato configurato un header http, l'indirizzo remoto sarà codificato in base64 se viene abilitata la seguente proprietà;
- govway-proxy-query: se configurato verrà utilizzato un parametro della url, con il nome indicato, per inoltrare al proxy l'indirizzo remoto;
- govway-proxy-query-base64: nel caso sia stato configurato un parametro della url, l'indirizzo remoto sarà codificato in base64 se viene abilitata la seguente proprietà.

È inoltre configurabile l'indicazione (true/false) se la funzionalità proxy deve essere attivata anche verso gli endpoint registrati nelle token policy e nelle attribute authority, tramite le seguenti proprietà:

- govway-proxy-token-introspection: servizio "introspection" definito in una *Token Policy Validazione*;
- govway-proxy-token-userinfo: servizio "user-info" definito in una *Token Policy Validazione*;
- govway-proxy-token-retrieve: *Token Policy Negoziazione*;
- govway-proxy-attribute-authority: *Attribute Authority* da cui vengono recuperati gli attributi.

Nota:

La configurazione dei parametri che riguardano l'header http o il parametro della url non sono obbligatori e se non presenti viene utilizzata la configurazione di default (header http "GovWay-APIAddress" non codificato in base64) ridefinibile nel file di configurazione locale "/etc/govway/govway_local.properties" tramite una configurazione come quella riportata di seguito (assumendo sia /etc/govway la directory di configurazione indicata in fase di installazione). Analogo discorso vale per l'attivazione della funzionalità proxy verso gli endpoint registrati nelle token policy e nelle attribute authority, la quale è per default disabilitata.

```
# ================================================
#  GovWay Proxy
#
# Default behaviour
org.openspcoop2.pdd.connettori.govwayProxy.enable=false
#
# Default configuration (HTTP)
org.openspcoop2.pdd.connettori.govwayProxy.header.enable=true
org.openspcoop2.pdd.connettori.govwayProxy.header.nome=GovWay-APIAddress
org.openspcoop2.pdd.connettori.govwayProxy.header.base64=false
#
# Default configuration (query URL)
org.openspcoop2.pdd.connettori.govwayProxy.urlParameter.enable=false
org.openspcoop2.pdd.connettori.govwayProxy.urlParameter.nome=govway_api_address
org.openspcoop2.pdd.connettori.govwayProxy.urlParameter.base64=false
#
# Default configuration (Token e Attributes)
org.openspcoop2.pdd.connettori.govwayProxy.tokenIntrospection.enable=false
org.openspcoop2.pdd.connettori.govwayProxy.tokenUserInfo.enable=false
org.openspcoop2.pdd.connettori.govwayProxy.tokenRetrieve.enable=false
org.openspcoop2.pdd.connettori.govwayProxy.attributeAuthority.enable=false
# ================================================
```

Nota: Anche l'abilitazione stessa della funzionalità "govway-proxy" può essere effettuata nel file di configurazione locale tramite la proprietà "org.openspcoop2.pdd.connettori.govwayProxy.enable" ed in questo caso non è necessario registrare la proprietà di sistema "govway-proxy-enable"

L'endpoint utilizzato per il proxy, indicato nella proprietà "govway-proxy", può essere ridefinito tramite le seguenti proprietà dalla più generica fino alla più specifica:

- govway-<ruolo>-proxy: l'endpoint indicato verrà utilizzato solamente se govway agisce nel ruolo indicato; "<ruolo>" può assumere i valori "fruizioni" o "erogazioni".
- profilo-<profilo>-govway-proxy o profilo-<profilo>-govway-<ruolo>-proxy: rispetto alla precedente proprietà è possibile restringere l'utilizzo dell'endpoint ad un determinato Profilo di Interoperabilità; "<profilo>" può assumere i valori "trasparente" (Profilo API Gateway), "modipa" (Profilo ModI), "spcoop" (Profilo SPCoop), "as4" (Profilo eDelivery), "sdi" (Profilo Fatturazione Elettronica).
- dominio-<nomeSoggetto>-govway-proxy o dominio-<nomeSoggetto>-govway-<ruolo>-proxy: l'endpoint indicato verrà utilizzato solamente per il soggetto interno indicato in "<nomeSoggetto>".
- dominio-<profilo>-<nomeSoggetto>-govway-proxy o dominio-<profilo>-<nomeSoggetto>-govway-<ruolo>-proxy: rispetto alla precedente proprietà è possibile restringere l'utilizzo dell'endpoint per il soggetto interno indicato in "<nomeSoggetto>" relativamente al solo Profilo di Interoperabilità indicato in "<profilo>".
- dominio-<tipoSoggetto>-<nomeSoggetto>-govway-proxy o dominio-<tipoSoggetto>-<nomeSoggetto>-govway-<ruolo>-proxy: rispetto alle precedenti due proprietà è possibile restringere l'utilizzo dell'endpoint per il soggetto interno indicato in "<nomeSoggetto>" relativamente al solo tipo indicato in "<tipoSoggetto>". Questa opzione è utile nei profili di interoperabilità dove ai soggetti è possibile associare più tipi, come ad es. in SPCoop dove sono utilizzabili i tipi "spc", "aoo", "test".
- tag-<nomeTag>-govway-proxy o tag-<nomeTag>-govway-<ruolo>-proxy: l'endpoint indicato verrà utilizzato solamente se l'API appartiene al tag indicato in "<nomeTag>".

Anche i parametri di configurazione relativamente all'utilizzo dell'header, al parametro della url possono essere ridefiniti, quando viene ridefinito un endpoint, con lo stesso criterio. Analogo discorso vale per l'attivazione della funzionalità proxy verso gli endpoint registrati nelle token policy e nelle attribute authority.

Nella figura Fig. 10.22 viene fornito un esempio di configurazione di un proxy relativamente alle sole fruizioni. L'endpoint del proxy è lo stesso per tutti i soggetti interni gestiti (dove è stato abilitato il multi-tenant) con la sola differenza che nel contesto della url è presente il nome del soggetto interno. In questo esempio l'endpoint remoto viene inserito nell'header HTTP GovWay-APIAddress codificato in base64.

Nella figura Fig. 10.23 viene fornito un esempio di configurazione di un proxy relativamente alle sole fruizioni dove l'endpoint del proxy differisce sulla porta a seconda del soggetto interno.

10.13 Autenticazione e Autorizzazione Principal (Security Constraint)

In precedenza, relativamente alla configurazione del controllo degli accessi, ed in particolare del meccanismo di autenticazione, si è indicata anche la possibilità di utilizzare il tipo *principal*. Questa configurazione richiede che l'autenticazione sia delegata all'application server o qualunque altra modalità che permetta a GovWay di accedere al principal tramite la api *HttpServletRequest.getUserPrincipal()*.

In precedenza, relativamente all'autorizzazione, si è descritta la possibilità di utilizzare ruoli con fonte *esterna*. Questa fonte richiede che la gestione dei ruoli sia delegata all'Application Server o a qualunque altra modalità che permetta a GovWay di accedere ai ruoli tramite la api *HttpServletRequest.isUserInRole()*.

Le modalità di configurazione di utenti e ruoli sull'application server variano in funzione della versione utilizzata e pertanto si rimanda alla documentazione del prodotto.

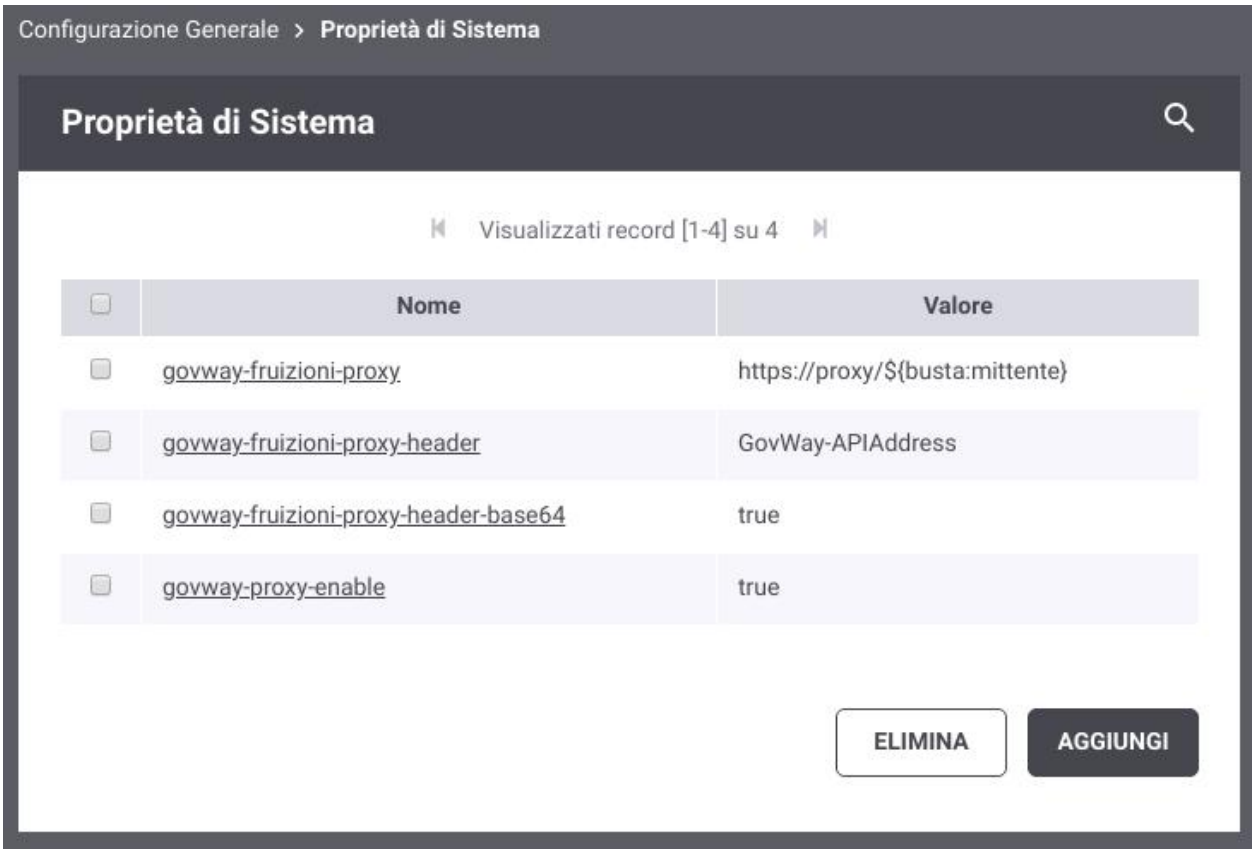


Fig. 10.22: GovWay Proxy per le fruizioni con endpoint dinamico

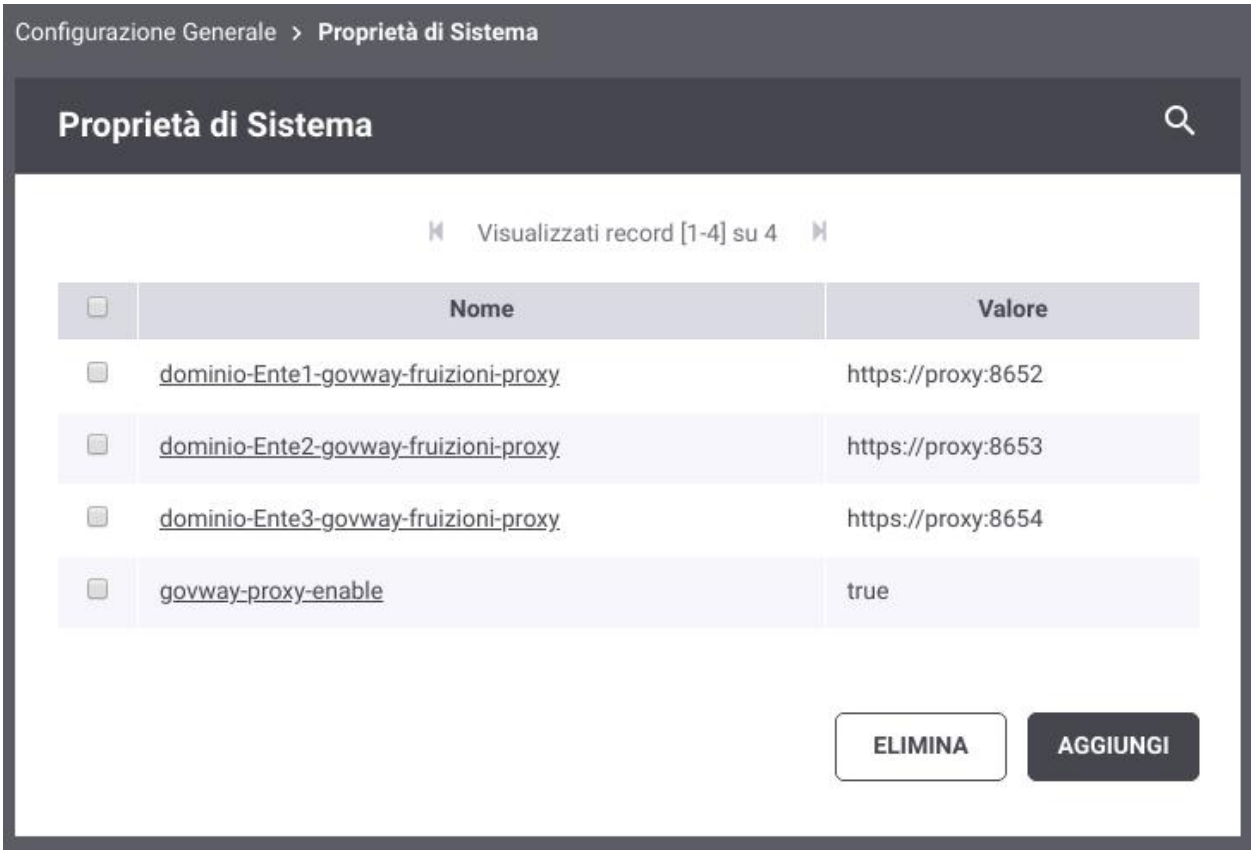


Fig. 10.23: GovWay Proxy per le fruizioni con endpoint differente per Soggetto Interno

È inoltre richiesto che l'applicazione utente sia protetta tramite un *security-constraint*. A tale scopo l'installazione di GovWay dispone di un contesto built-in *govwaySec* (definito nel war govwaySec.war/WEB-INF/web.xml) protetto tramite security constraint con metodo di autenticazione *HTTP-BASIC*:

```
<security-constraint>
        <web-resource-collection>
                <web-resource-name>AuthenticationContainer</web-resource-name>
                <url-pattern>/*</url-pattern>
        </web-resource-collection>
        <auth-constraint>
                <role-name>*</role-name>
        </auth-constraint>
</security-constraint>

...

<security-role>
    <role-name>*</role-name>
</security-role>

<login-config>
    <auth-method>BASIC</auth-method>
</login-config>
```

Se si intende utilizzare una configurazione differente di quella built-in si deve procedere con la modifica di tale descrittore web.xml presente all'interno dell'archivio.

Nota: Con questo tipo di configurazione, le URL che gli applicativi devono invocare devono essere adeguate sostituendo il contesto *govway* con *govwaySec*.

10.14 Espressioni XPath su messaggi JSON

In diverse funzionalità (*Correlazione Applicativa*, *Registrazione di una policy*, *Modalità di identificazione dell'azione*) è stata documentato la possibilità di utilizzare espressioni jsonPath o XPath per estrarre contenuti dai messaggi JSON o XML in transito sul Gateway.

L'estrazione dei contenuti da messaggi JSON si basa su espressioni JSONPath che allo stato attuale non hanno la stessa «potenza» delle espressioni XPath. Ad esempio:

- non è possibile ottenere il nome di un claim, come invece in XPath è possibile ottenere il local-name di un elemento tramite la funzione "local-name"
- non si dispongono delle complesse funzioni per le elaborazioni sulle stringhe (ad es. in xpath è disponibile la funzione "substring-before")
- ...

Per ovviare a tali limitazioni GovWay fornisce la possibilità di utilizzare espressioni XPath su messaggi JSON attraverso la seguente sintassi:

```
xpath [namespace(prefix1:uri1, ... ,prefixN:uriN) ] <espressioneXPathStandard>
```

Nel caso il gateway rilevi una espressione che inizi con il prefisso "xpath " da applicare su un messaggio JSON, effettua una trasformazione del messaggio in una rappresentazione xml. Ad esempio per il messaggio JSON:

```
{
        "prova":"test1",
        "prova2":23
}
```

Per estrarre il valore del field "prova" è possibile utilizzare le seguenti espressioni, la prima jsonPath e le successive xpath:

- $.prova
- xpath //prova/text()
- xpath /json2xml/prova/text()

La espressioni xpath sono utilizzabili poichè il messaggio JSON viene convertito nel seguente messaggio xml (inserito all'interno dell'elemento radice "json2xml"):

```
<json2xml>
        <prova>test1</prova>
        <prova2>23</prova2>
</json2xml>
```

Mentre nell'esempio precedente sono sufficienti le funzionalità offerte dal jsonPath per estrarre il valore del field "prova", ricorrere all'utilizzo di xPath è necessario se ad esempio vogliamo ottenere il nome di un field. Nell'esempio seguente l'espressione fornita consente di estrarre il nome dell'ultimo field presente nella struttura json "prova2". Tale risultato è ottenibile solamente utilizzando l'espressione xPath:

```
xpath local-name(/json2xml/*[last()])
```

In alcuni contesti i servizi REST non vengono implementati a partire da interfacce progettate ad hoc (OpenAPI, Swagger . . .) ma sono frutto di una trasformazioni di esistenti servizi SOAP. In questi scenari, i servizi REST veicolano messaggi JSON ottenuti attraverso la trasformazione dei relativi messaggi XML utilizzati su SOAP. Per poter utilizzare espressioni xPath devono essere affrontate le problematiche di risoluzione dei prefissi e dei namespace. In questi contesti i messaggi JSON presenteranno field che possiedono nel nome il carattere ":" ereditato dalla rappresentazione xml. Di seguito un esempio di messaggio json ottenuto da una trasformazione di un messaggio xml equivalente:

```
{
    "m:NomeAzioneTestRequest":  {
        "bodyWithNS" : "true",
        "xmlns:m" : "http://testNamespace",
        "prodotto" : {
            "codice" : "26",
            "altro:codice3" : "34",
            "xmlns:altro" : "http://testNamespaceAltro"
        }
    }
}
```

Supponendo di voler estrarre il nome del field "NomeAzioneTestRequest" e da questo eliminare anche il suffisso "Request" è possibile utilizzare la seguente espressione xPath:

```
xpath namespace(m:http://testNamespace, altro:http://altro) substring-before(local-
↪name(//json2xml/*),\"Request\")
```

Si può notare come tra il prefisso "xpath " e l'espressione xpath vera e propria (substring-before(. . .)) siano stati definiti i namespace che coinvolgono i field presenti nella struttura json che avevano il carattere ":".

La struttura xml, ottenuta dalla conversione del messaggio json, su cui viene applicata l'espressione xpath è la seguente:

```
<json2xml xmlns:m="http://testNamespace"  xmlns:altro="http://altro"  xmlns:___xmlns=
↪"http://govway.org/utils/json2xml/xmlns">
        <m:NomeAzioneTestRequest>
                <bodyWithNS>true</bodyWithNS>
                <___xmlns:m>http://testNamespace</___xmlns:m>
                <prodotto>
                        <codice>26</codice>
                        <altro:codice3>34</altro:codice3>
                        <___xmlns:altro>http://testNamespaceAltro</___xmlns:altro>
                </prodotto>
        </m:NomeAzioneTestRequest>
</json2xml>
```

Nota: Il prefisso "xmlns:" viene gestito automaticamente da GovWay, il quale gli associa un namespace di default "http://govway.org/utils/json2xml/xmlns". Tale namespace è possibile ridefinirlo aggiungendo all'elenco dei namespace anche un mapping per "xmlns".

10.15 Validazione dei messaggi con OpenAPI 3.x

Nella sezione *Validazione dei messaggi* è stata descritta la funzionalità di validazione dei messaggi applicativi in transito sul gateway.

Dalla versione 3.3.1, per la validazione dei messaggi riguardanti API REST con specifiche di interfaccia OpenAPI 3.x, viene utilizzata la libreria openapi4j (https://openapi4j.github.io/openapi4j/). È possibile ritornare al precedente motore di validazione registrando la seguente *Proprietà* sull'erogazione o sulla fruizione:

- *validation.openApi4j.enabled=false*

Dalla versione 3.3.5.p1 è inoltre possibile utilizzare un ulteriore motore di validazione, utilizzando la libreria "swagger-request-validator" (https://bitbucket.org/atlassian/swagger-request-validator). È possibile attivare il nuovo motore di validazione registrando la seguente *Proprietà* sull'erogazione o sulla fruizione:

- *validation.swaggerRequestValidator.enabled=true*

Se invece si vuole modificare il tipo di validazione effettuata con i motori "openapi4j" o "swagger-request-validator" è possibile farlo abilitando (true) o disabilitando (false) la specifica funzionalità registrando una delle seguenti *Proprietà* (per default tutte le proprietà elencate sono abilitate):

- *validation.openApi.validateAPISpec* (default: true): prima di procedere con la validazione del messaggio, viene controllato che l'interfaccia OpenAPI 3.x sia sintatticamente valida;
- *validation.openApi.validateRequestQuery* (default: true): viene effettuata la validazione della query url;
- *validation.openApi.validateRequestHeaders* (default: true): viene effettuata la validazione degli header http della richiesta;
- *validation.openApi.validateResponseHeaders* (default: true): viene effettuata la validazione degli header http della risposta;
- *validation.openApi.validateRequestCookies* (default: true): viene effettuata la validazione dei cookie presenti nella richiesta;
- *validation.openApi.validateRequestBody* (default: true): viene effettuata la validazione del payload http della richiesta;
- *validation.openApi.validateResponseBody* (default: true): viene effettuata la validazione del payload http della risposta;

- *validation.openApi.mergeAPISpec* (default: true): eventuali schemi esterni json o yaml vengono aggiunti all'OpenAPI principale prima di procedere con la validazione.

Per il motore di validazione "swagger-request-validator" sono disponibili le ultiori proprietà:

- *validation.swaggerRequestValidator.validateWildcardSubtypeAsJson* (default: true): consente di indicare se le richieste associate a media type definiti con il carattere "*" nel subtype (es. application/*) debbano essere validate come richieste json;
- *validation.swaggerRequestValidator.validateRequestUnexpectedQueryParam* (default: false): se abilitata vengono segnalati gli eventuali parametri non definiti nella specifica;
- *validation.swaggerRequestValidator.resolveFullyApiSpec* (default: false): indica se sostituire inline i $ref nello schema con le loro definizioni. Per default viene utilizzato il valore "false" poichè quando vengono risolti inline non vengono gestiti correttamente i singoli attributi degli schemi combinati (oneOf, allOf ecc..). La risoluzione inline consente però di avere delle performance maggiori.
- *validation.swaggerRequestValidator.injectingAdditionalPropertiesFalse* (default: false): se abilitata, viene riattivato il transformer della libreria che aggiunge additionalProperties=false in tutti gli oggetti degli schemi. È necessario disattivarlo per poter validare correttamente gli schemi che definiscono tale proprietà a true. La libreria lo utilizza come workaround per validare strutture allOf.

10.16 Cifratura delle Password

Gli oggetti censiti nel registro di GovWay che possiedono una password sono i seguenti:

- le utenze delle console di gestione e monitoraggio (descritte nella sezione *Utenti*);
- gli applicativi e i soggetti registrati con credenziali "http-basic" (sezione *Credenziali "http-basic"*);
- gli applicativi e i soggetti registrati con credenziali "api-key"; in questo caso viene cifrata la chiave di identificazione univoca (sezione *Credenziali "api-key"*).

Le password vengono cifrate per default con un algoritmo di cifratura: SHA-512-based Unix crypt (6).

Nota: Per garantire la retrocompatibilità con le utenze esistenti precedenti alla versione 3.3.2 di GovWay, la verifica delle password viene attuata anche usando il precedente algoritmo. La verifica in modalità "backward compatibility" può essere disattivata, una volta migrate tutte le password al nuovo formato di cifratura, agendo sul file <directory-lavoro>/consolePassword.properties:

```
# Abilitare l'opzione seguente per poter autenticare:
# - le utenze delle console esistenti memorizzate con la precedente␣
↪cifratura MD5
# - le password 'basic' degli applicativi/soggetti memorizzati in chiaro
passwordEncrypt.backwardCompatibility=false
```

È possibile modificare il tipo di cifratura configurando i parametri presenti nel file <directory-lavoro>/consolePassword.properties:

```
# Tipo di cifratura (enum org.openspcoop2.utils.crypt.CryptType)
passwordEncrypt.type=SHA2_BASED_UNIX_CRYPT_SHA512
# In alternativa alla definizione di un tipo, è possibile fornire una classe␣
↪che implementa l'interfaccia org.openspcoop2.utils.crypt.ICrypt
#passwordEncrypt.customType=className

# Charset utilizzato per le password
```

(continues on next page)

(continua dalla pagina precedente)

```
#passwordEncrypt.charsetName=UTF-8

# Parametri per il calcolo del 'salt'
passwordEncrypt.salt.length=16
passwordEncrypt.salt.secureRandom=true
#passwordEncrypt.salt.secureRandomAlgorithm=SHA1PRNG

# Parametri per il calcolo del Digest
#passwordEncrypt.digestAlgorithm=
#passwordEncrypt.iteration=intNumber

# Output format
#passwordEncrypt.base64Encoding=true/false
```

I tipi di cifratura supportati sono:

- *SHA2_BASED_UNIX_CRYPT_SHA256* e *SHA2_BASED_UNIX_CRYPT_SHA512*: SHA2-based Unix crypt in variante SHA-256 e SHA-512; consentono la personalizzazione del "salt" e del numero di iterazioni ("rounds").
- *LIBC_CRYPT_MD5* e *LIBC_CRYPT_MD5_APACHE*: libc crypt() MD5 «1» e variante Apache «$apr1$»; consentono la personalizzazione del "salt".
- *DES_UNIX_CRYPT*: Unix crypt(3) DES; consente la personalizzazione del "salt".
- *RFC2307_MD5*, *RFC2307_SMD5*, *RFC2307_SHA* e *RFC2307_SSHA*: RFC2307 in variante MD5, SMD5, SHA e SSHA; non consente alcuna personalizzazione.
- *JASYPT_BASIC_PASSWORD* e *JASYPT_STRONG_PASSWORD*: Jasypt in variante basic e strong; non consente alcuna personalizzazione.
- *JASYPT_CUSTOM_PASSWORD*: Jasypt custom configurabile per "salt", numero di iterazioni, algoritmo di digest e codifica base64/hex.
- *PBE_KEY_SPEC*: PBE Key Spec configurabile per "salt", numero di iterazioni, algoritmo di digest e codifica base64/hex.
- *B_CRYPT* e *S_CRYPT*: BCrypt e SCrypt; non consente alcuna personalizzazione.
- *PLAIN*: le password vengono salvate in chiaro