
Scenari Applicativi

Release 3.3.10

Link.it

20 gen 2023

1	Ambiente di esecuzione	1
1.1	Prerequisiti	1
1.2	Avvio Ambiente	2
1.3	Progetto Postman	4
2	Profilo “API Gateway”	11
2.1	Erogazione pubblica	11
2.2	Erogazione OAuth	15
3	Profilo “ModI”	23
3.1	Pattern “ID_AUTH”	24
3.2	Pattern “INTEGRITY”	54
3.3	Pattern “ID_AUTH” via PDND	83
3.4	Pattern “ID_AUTH” via PDND + “INTEGRITY”	122
4	Monitoraggio	155
4.1	Transazione in errore	155
4.2	Transazione con esito corretto	159

Ambiente di esecuzione

Per semplificare la realizzazione e la verifica degli scenari d'uso, descritti in questa sezione della documentazione di Govway, è possibile dotarsi dell'ambiente di esecuzione appositamente predisposto.

Nella sezione *Prerequisiti* vengono indicati i software di base richiesti per poter avviare l'ambiente e verificare gli scenari.

Indicazioni su come ottenere un ambiente, preconfigurato per verificare gli scenari, sono presenti nella sezione *Avvio Ambiente*.

Infine nella sezione *Progetto Postman* vengono fornite indicazioni su come ottenere un progetto Postman che contenga i client preconfigurati per attuare le richieste descritte in ogni scenario.

1.1 Prerequisiti

Per l'avvio dell'ambiente di esecuzione degli scenari è necessario disporre del seguente software di base:

- dotarsi di una installazione **Docker** che gestirà l'intero contesto di esecuzione degli scenari;
- dotarsi dell'applicativo **Postman** utilizzato come client per l'invio delle richieste a Govway.

L'ambiente di esecuzione è composto da:

- **ambiente docker-compose** preinizializzato con gli scenari descritti in questo manuale;
- **progetto Postman** preconfigurato per verificare gli scenari:
 - invocazione pubblica o OAuth su profilo “API Gateway”;
 - profilo “Modl” su API REST;
 - profilo “Modl” su API SOAP.

Gli scenari configurati sull'ambiente docker devono poter accedere ai seguenti servizi su internet:

- Petstore: <https://petstore.swagger.io/>
- Credit Card Verification: <https://ws.cdyne.com/creditcardverify/luhnchecker.asmx>

1.2 Avvio Ambiente

Dopo aver scompattato l’archivio, indicato nei prerequisiti, sarà possibile avviare un ambiente tramite docker compose preinizializzato per gli scenari descritti nel manuale. Di seguito vengono forniti tutti i passaggi da effettuare per ottenere un ambiente funzionante:

- *Archivio*: scompattare l’archivio nella cartella di destinazione scelta per ospitare l’ambiente di esecuzione degli scenari.
- *Hostname*: l’ambiente è configurato per utilizzare l’hostname “govway.localdomain”. Configurare una risoluzione dell’hostname ad esempio registrando nel file /etc/hosts l’entry:

127.0.0.1	govway.localdomain
-----------	--------------------

- *Ambiente Docker*: avviare l’ambiente docker compose utilizzando lo script “starttest.sh” presente all’interno della cartella di destinazione dell’ambiente (Fig. 1.1).

```
[root@poli-nb18 AmbienteDocker]# ./starttest.sh
Starting goauth ...
Starting spid_testenv ...
Starting goauth
Starting ambientedocker_init_1 ...
Starting ambientedocker_init_1
Starting ambientedocker_init_1 ... done
Starting PGSQ95 ...
Starting gatewaystenv ... done
Starting PGSQ95 ... done
Starting keycloak ...
Starting keycloak ... done
Starting traefik ...
Starting traefik ... done
```

Fig. 1.1: Schermata di avvio «docker-compose up»

I componenti avviati sono i seguenti:

- gateway: l’istanza di Govway
- PGSQ95: il database Postgres
- keycloak: l’authorization server
- traefik: il load balancer

Nota: Lo script “starttest.sh” si occupa di inizializzare due variabili di ambiente prima di avviare l’ambiente tramite il comando “*docker-compose up*”:

- SERVER_FQDN: definisce l’hostname dell’ambiente (negli esempi govway.localdomain)
- LOCAL_DATA: directory contenente gli storage locali utilizzate dalle immagini docker avviate dal compose (l’archivio fornisce già la directory ./data)

Dopo aver avviato l'ambiente è possibile verificare l'accesso alle seguenti console:

- *GovWay - Console di Gestione*: permette di visualizzare le configurazioni realizzate su Govway (Fig. 1.2).



Fig. 1.2: Accesso alla console di gestione

- *GovWay - Console di Monitoraggio*: permette di consultare le transazioni gestite da Govway (Fig. 1.3).

```
endpoint: https://govway.localdomain/govwayMonitor/
username: operatore
password: 123456
```

- *Keycloak - Authorization Server*: permette di consultare le configurazioni realizzate sull'Authorization Server Keycloak (Fig. 1.4).

```
endpoint: https://govway.localdomain/auth/
username: admin
password: admin
```

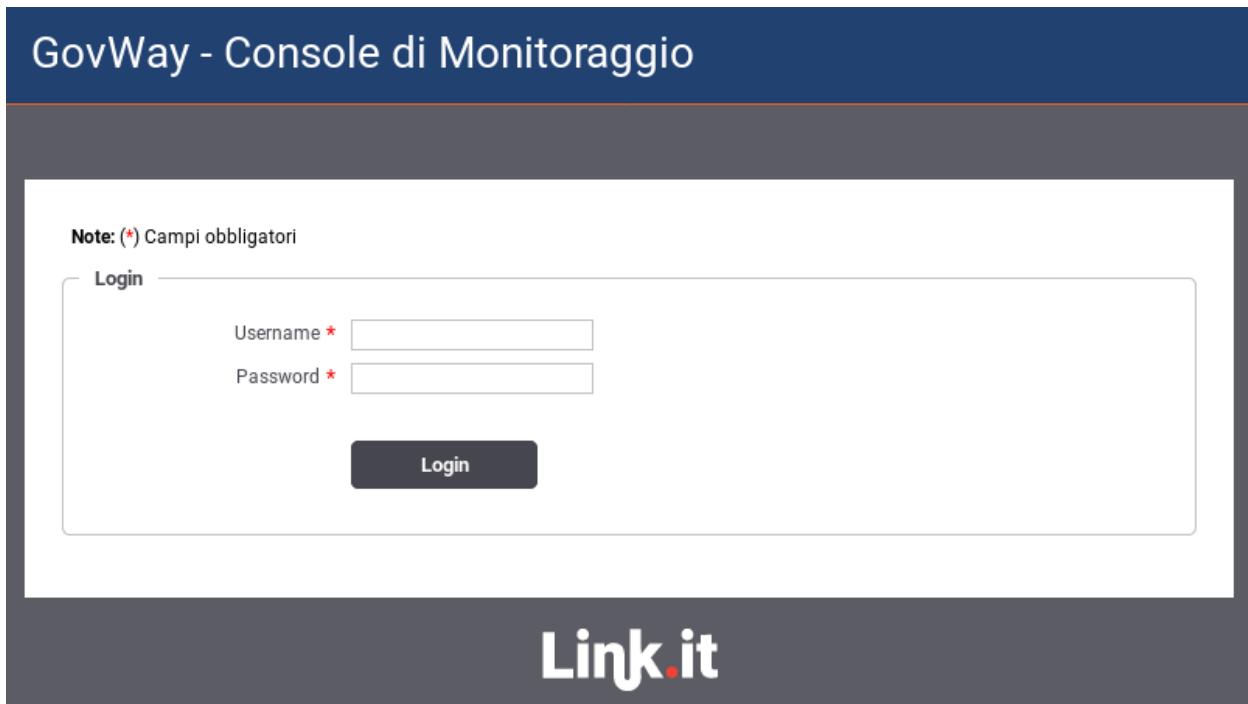


Fig. 1.3: Accesso alla console di monitoraggio

1.3 Progetto Postman

La collezione Postman comprende tutte le configurazioni utilizzate nei vari scenari presentati (Fig. 1.5). La collection deve essere caricata sul proprio Postman tramite la funzionalità di import.

Una volta effettuato il caricamento della collezione, modificare i parametri della collezione (Fig. 1.6) al fine di indicare nella variabile “*hostname*” (Fig. 1.7) l’indirizzo ip su cui è stato attivato l’immagine docker compose (per default è presente 127.0.0.1).

Infine accedere alla configurazione generale di Postman (Fig. 1.8) ed assicurarsi che la voce “*SSL Certificate Verification*” nella maschera “*General*” sia disabilitata (Fig. 1.9) e che non vi sia impostato un proxy nella maschera “*Proxy*” (Fig. 1.10).

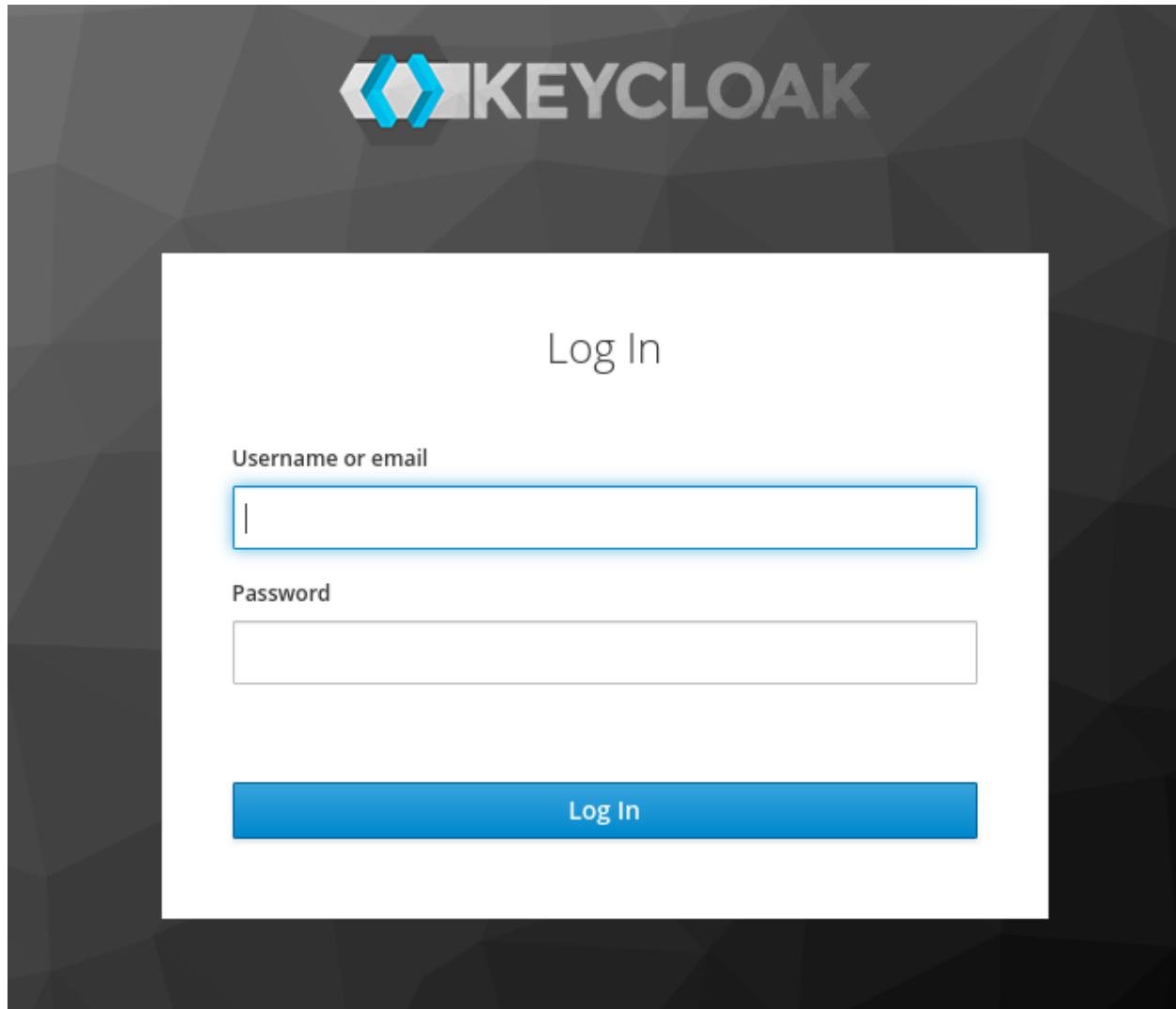


Fig. 1.4: Accesso alla console dell'authorization server

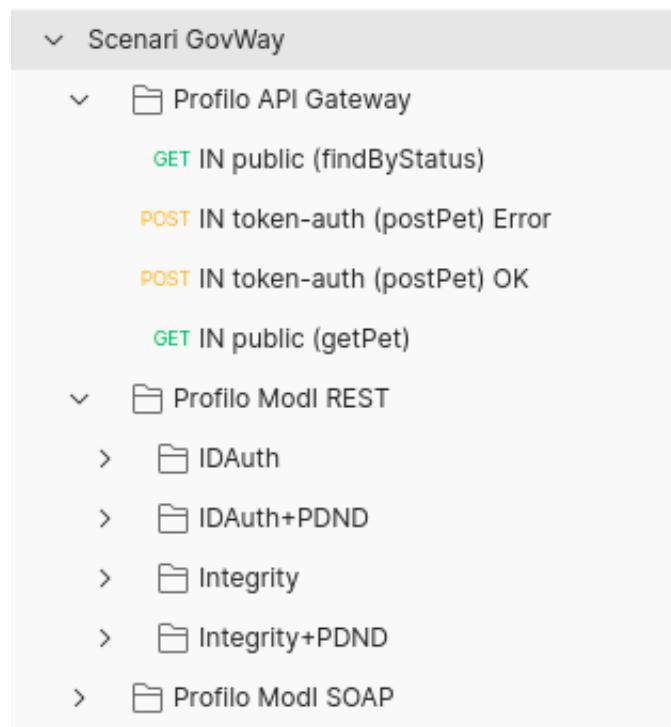


Fig. 1.5: Indice della collection Postman

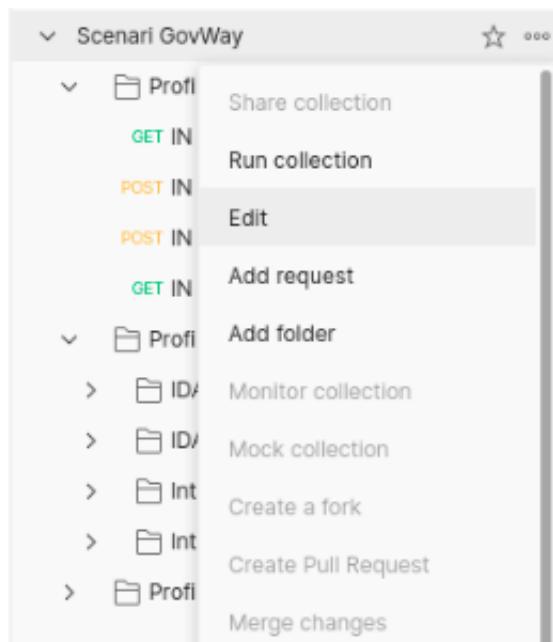


Fig. 1.6: Configurazione Collection Postman

EDIT COLLECTION X

Name
Scenari GovWay

Description Authorization Pre-request Scripts Tests **Variables** ●

These variables are specific to this collection and its requests. [Learn more about collection variables.](#)

	VARIABLE	INITIAL VALUE i	CURRENT VALUE i	...	Persist All	Reset All
<input checked="" type="checkbox"/>	hostname	127.0.0.1	127.0.0.1			
<input checked="" type="checkbox"/>	govway-url	https://{{hostname}}/go...	https://{{hostname}}/govway			
<input checked="" type="checkbox"/>	soggetto	Ente	Ente			
<input checked="" type="checkbox"/>	soggettoEsterno	EnteEsterno	EnteEsterno			
<input checked="" type="checkbox"/>	keycloak-url-auth	https://{{hostname}}/aut...	https://{{hostname}}/auth/realms/master/protocol/openid-conn...			
<input checked="" type="checkbox"/>	keycloak-url-token	https://{{hostname}}/aut...	https://{{hostname}}/auth/realms/master/protocol/openid-conn...			
<input checked="" type="checkbox"/>	keycloak-client-id	oauth2-app1	oauth2-app1			
<input checked="" type="checkbox"/>	keycloak-client-secret	fd5f09fa-028d-461b-8e4f...	fd5f09fa-028d-461b-8e4f-063c111c069f			

i Use variables to reuse values in different places. Work with the current value of a variable to prevent sharing sensitive values with your team. [Learn more about variable values](#) X

Cancel Update

Fig. 1.7: Configurazione Hostname nella Collection Postman

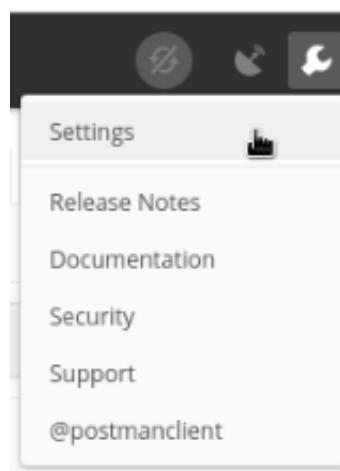


Fig. 1.8: Configurazione Generale Postman

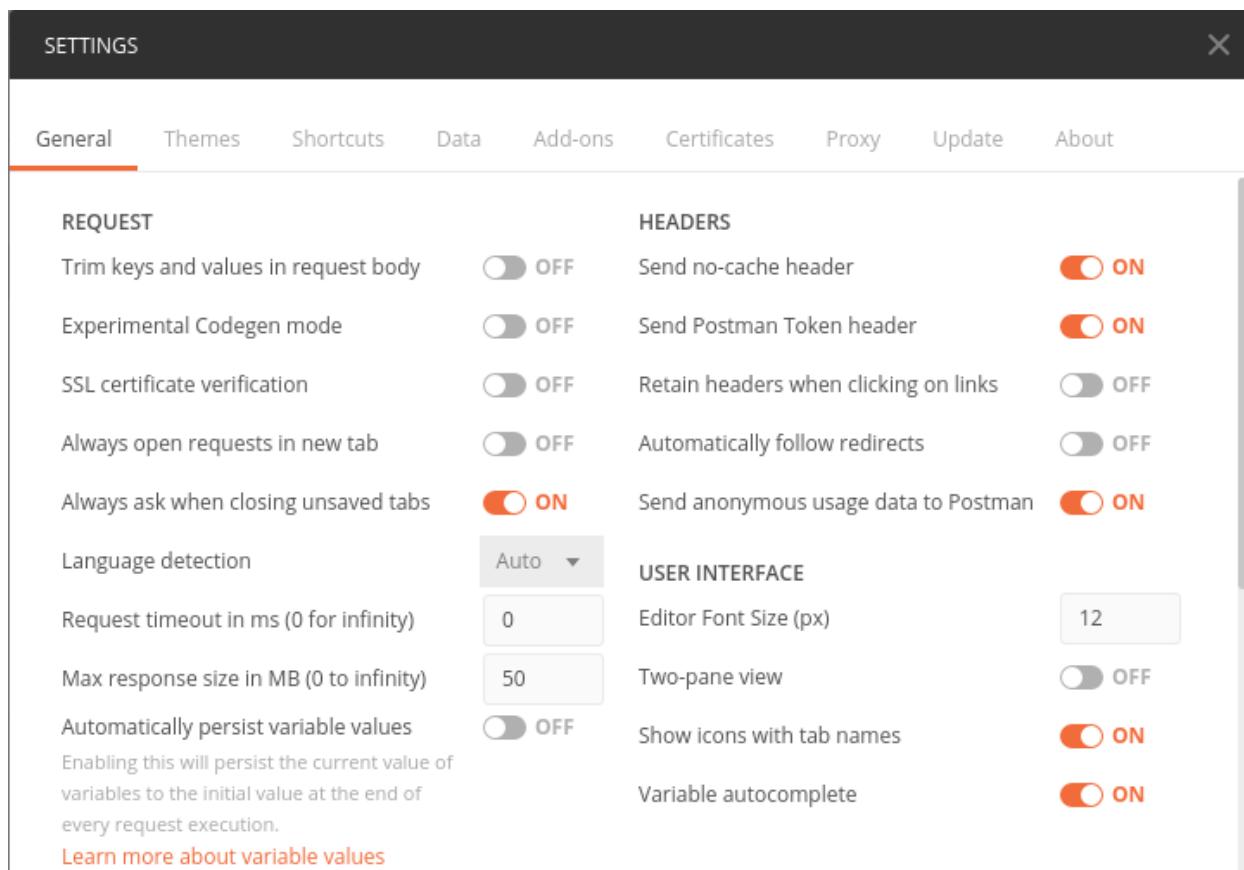


Fig. 1.9: Configurazione SSL Postman

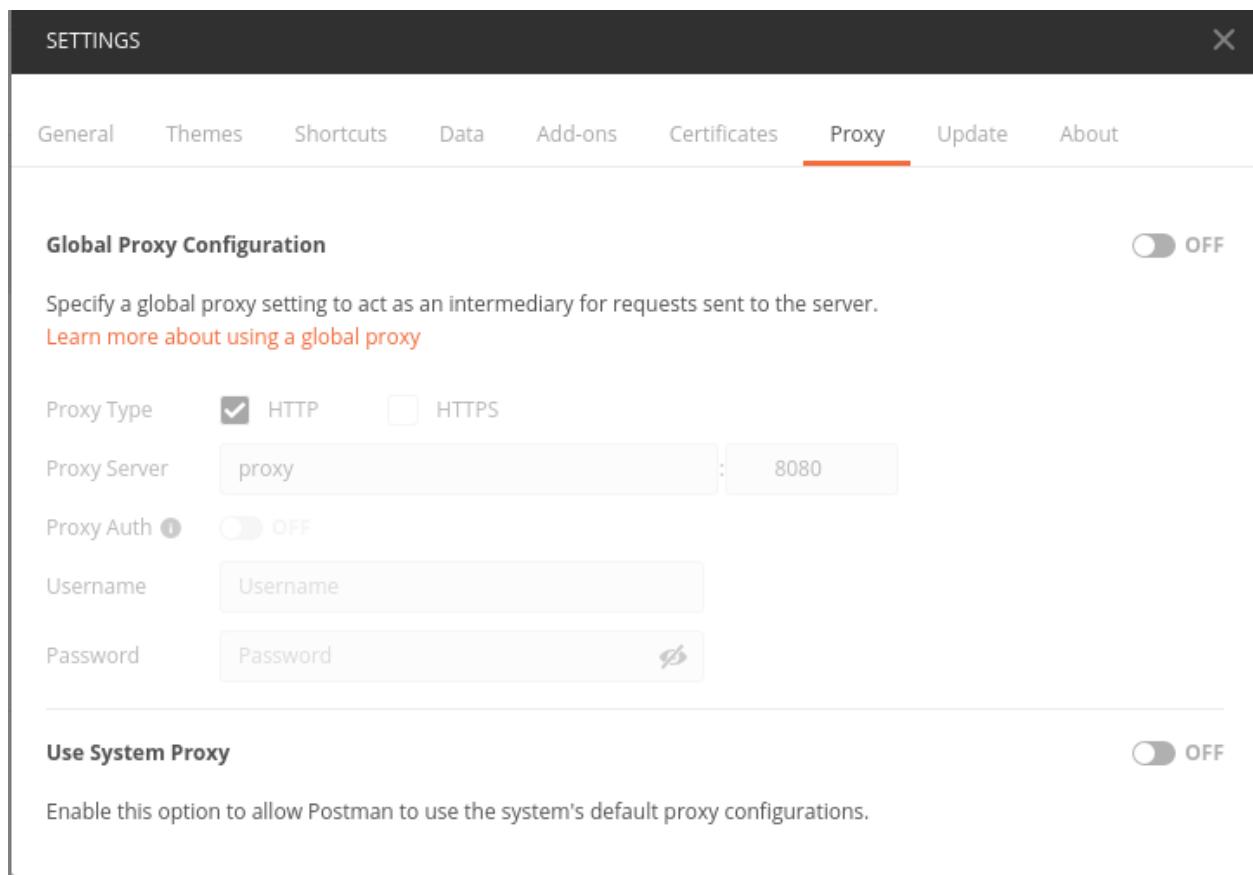


Fig. 1.10: Configurazione Proxy Postman

CAPITOLO 2

Profilo “API Gateway”

Nelle sezioni successive verranno mostrati degli scenari di esempio di una API Rest erogata con profilo “API Gateway».

Nel primo scenario descritto la sua fruizione è a disposizione di qualsiasi client poichè non vi sono meccanismi di autenticazione/autorizzazione configurati.

Nel secondo scenario viene invece richiesto un token OAuth.

Nota: Per una consultazione mirata alle informazioni di interesse per lo scenario si consiglia di impostare nel menù in alto a destra il profilo “API Gateway” come mostrato nella figura Fig. 2.1.



Fig. 2.1: Selezione del profilo “API Gateway”

2.1 Erogazione pubblica

Obiettivo

Esporre tramite Govway un servizio con accesso pubblico (forma anonima).

Sintesi

In questo scenario è richiesta l'esposizione tramite gateway di un servizio da erogare, consentendo il libero accesso ai fruitori, che potranno invocare la relativa interfaccia senza presentare alcuna credenziale.

Per illustrare questo scenario, abbiamo scelto il servizio «PetStore», che sarà reso accessibile da Govway tramite l'interfaccia REST in versione OpenAPI 3.

La figura seguente descrive graficamente questo scenario.

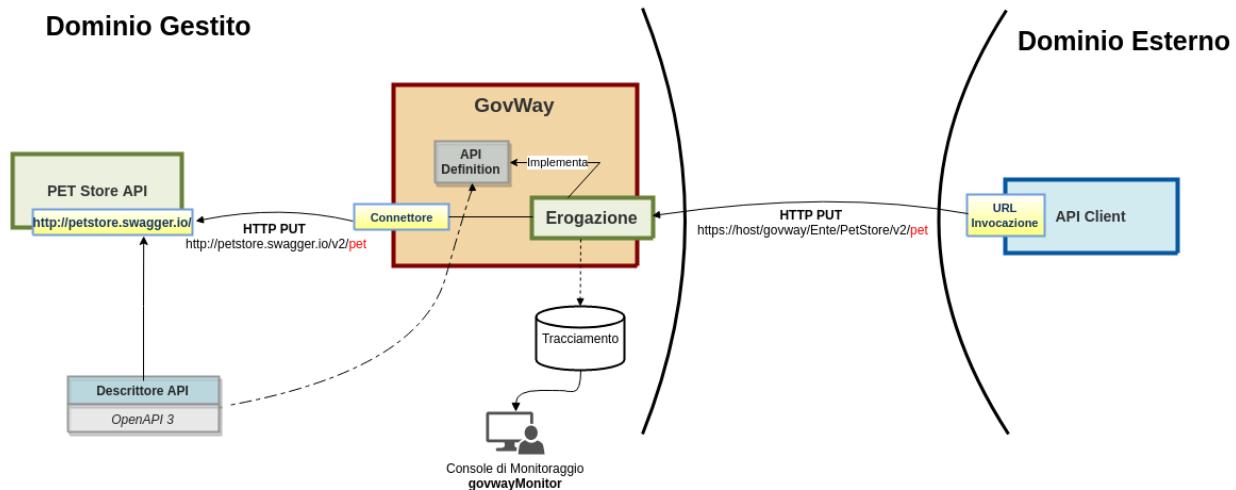


Fig. 2.2: Erogazione ad accesso pubblico

2.1.1 Esecuzione

I fruitori del servizio «PetStore» invocano le operazioni disponibili tramite i propri client senza utilizzare alcuna forma di autenticazione, utilizzando come “base-uri” la url di invocazione di GovWay

Avvalendosi del progetto Postman a corredo, eseguire «*IN public (findByStatus)*» per verificare l'esecuzione dell'erogazione del servizio PetStore con libero accesso.

2.1.2 Configurazione

In questa sezione vengono mostrate le parti di interesse relative alla configurazione con accesso pubblico.

Si assume che sia stata configurata una API "PetStore" con il descrittore OpenAPI 3 (scaricabile al seguente [indirizzo](#)).

Per registrare una erogazione dell'API "PetStore" pubblicamente accessibile si deve cliccare sul pulsante «Aggiungi» all'interno della sezione «Erogazione» (Fig. 2.5):

1. Selezionare l'API «PetStore v1» nel riquadro delle Informazioni Generali.
 2. Selezionare l'accesso API «pubblico» nel riquadro Controllo dei Accessi.
 3. Verificare che il campo «Endpoint», nel riquadro Connettore, sia stato correttamente inizializzato sulla base del valore di default presente nel descritto della API.

Nota: Verifica del certificato server

Poichè il servizio PetStore è disponibile solamente in https, modificare il prefisso dell'endpoint fornito. Inoltre per validare il certificato ritornato dal server “petstore.swagger.io” deve essere effettuata una opportuna configurazione del trustStore tls come descritto nella sezione avanzate_connettori_https. Poichè non è obiettivo di questo scenario si suggerisce di disabilitare la validazione del certificato server se si rilevano problematiche di trust del certificato server.

- #### 4. Salvare la configurazione dell'erogazione.

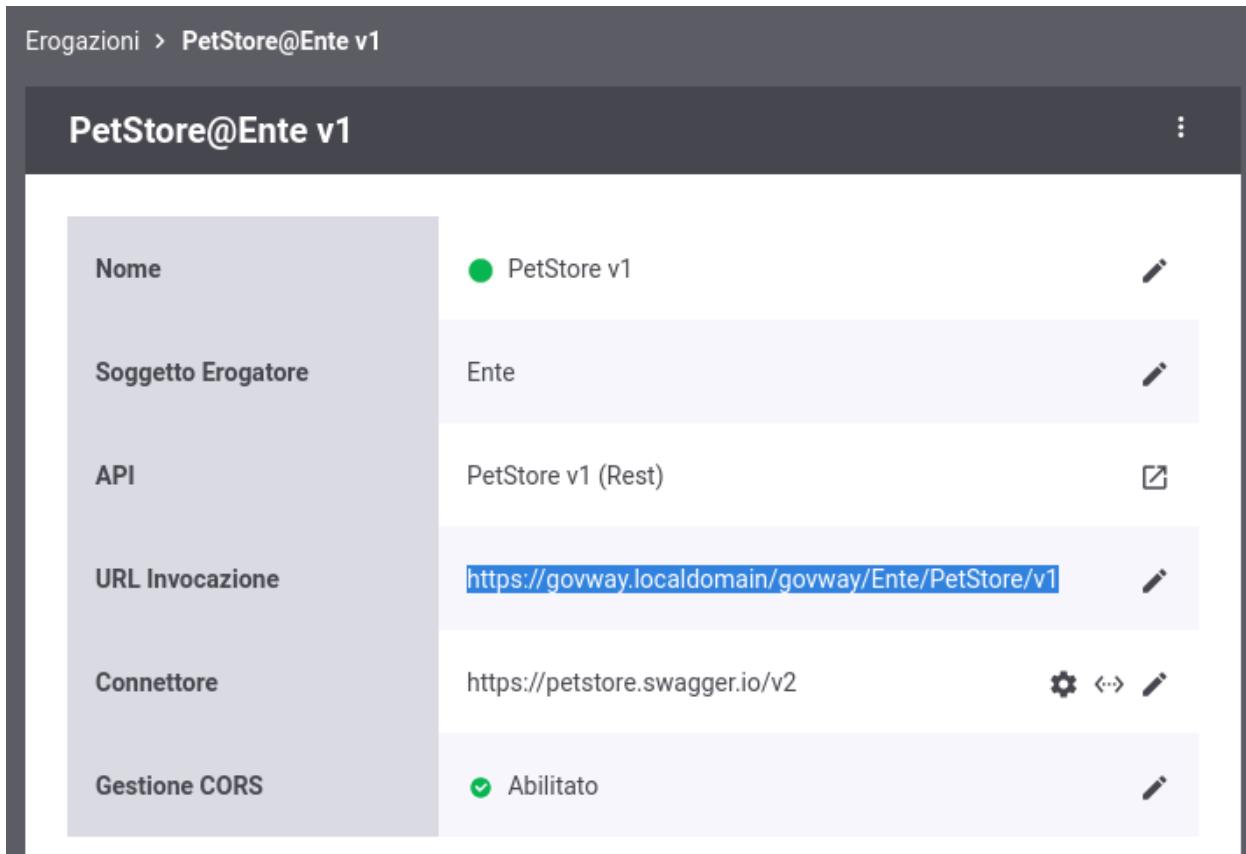


Fig. 2.3: Erogazione pubblica, url di invocazione

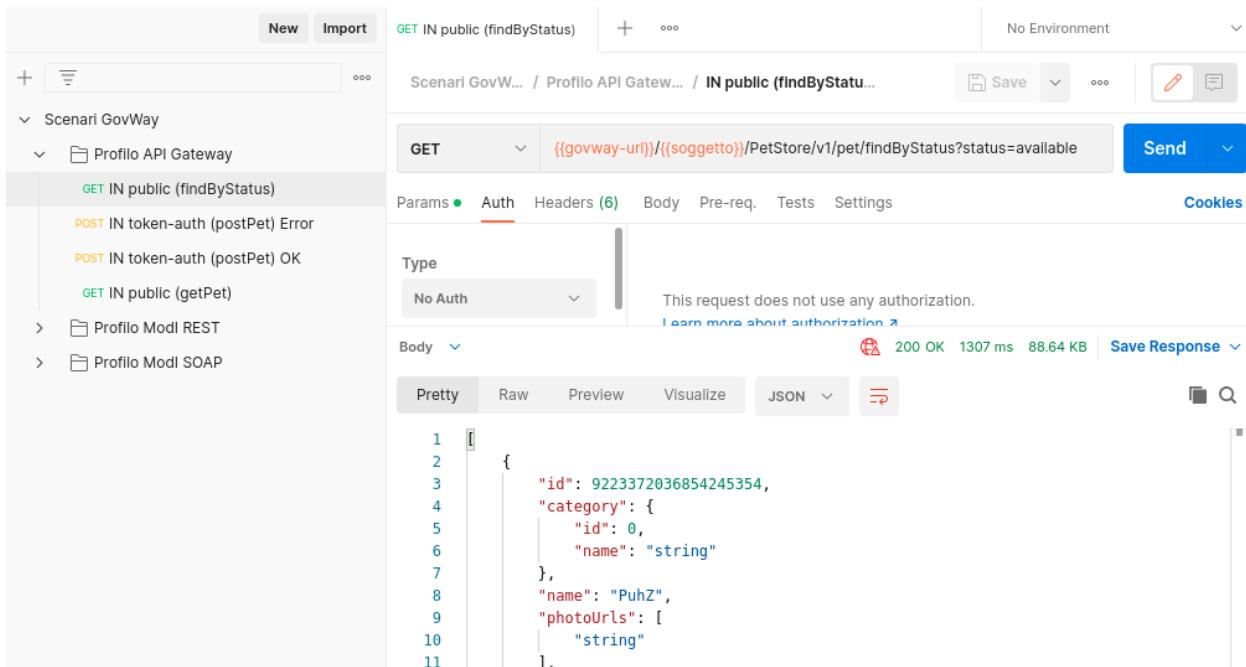


Fig. 2.4: Erogazione pubblica, esecuzione da Postman

Erogazioni > Aggiungi

Note: (*) Campi obbligatori

Informazioni Generali

API

Nome	PetStore v1
Tipo	Rest

Controllo degli Accessi

Accesso API	pubblico
-------------	----------

Connettore

Endpoint *	https://petstore.swagger.io/v2
Autenticazione Http	<input type="checkbox"/>
Autenticazione Token	<input type="checkbox"/>
AutenticazioneHttps	<input checked="" type="checkbox"/>
Proxy	<input type="checkbox"/>
Ridefinisci Tempi Risposta	<input type="checkbox"/>

AutenticazioneHttps

Tipologia	TLSv1.3
Verifica Hostname	<input checked="" type="checkbox"/>
Autenticazione Server	
Verifica	<input type="checkbox"/>
Autenticazione Client	
Abilitato	<input type="checkbox"/>

SALVA

5. Nel dettaglio della configurazione dell'erogazione è possibile vedere come non vi sia abilitato alcun controllo nella voce “Controllo Accessi”.

Nota: Esaminando l'erogazione preconfigurata si può notare come le risorse siano state suddivise in due gruppi in cui varia proprio il controllo degli accessi, e la risorsa invocata (GET /pet/findByStatus) rientra nel gruppo “Predefinito” dove il controllo degli accessi risulta disabilitato. L'altro gruppo verrà descritto nello scenario *Erogazione OAuth*.

Nome Gruppo	Predefinito
Elenco Risorse	GET /pet/findByStatus, GET /pet/findByTags, GET /pet/{petId}, GET /store/inventory, GET /store/order/{orderId}, GET /user/login, GET /user/logout, ...
Controllo Accessi	Disabilitato
Rate Limiting	Disabilitato
Validazione	Disabilitato

Fig. 2.6: Configurazione dell'erogazione

2.2 Erogazione OAuth

Obiettivo

Esporre un servizio accessibile tramite protocollo OAuth2 (Authorization Code).

Sintesi

Assumendo che sia stata effettuata la configurazione di un'erogazione ad accesso pubblico (vedi scenario *Erogazione pubblica*), verifichiamo in questo scenario come impostare il sistema di controllo degli accessi affinché il servizio richieda un token di sicurezza, come previsto dal protocollo OAuth2. In particolare la limitazione dell'accesso sarà configurata solo per le operazioni di scrittura, lasciando libero accesso per le letture.

La figura seguente descrive graficamente questo scenario.

I passi previsti sono i seguenti:

1. Il client entra in possesso del token, previa autenticazione e consenso dell'utente richiedente.
2. Il client utilizza il token per l'invio della richiesta.
3. Govway valida il token ricevuto e verifica i criteri di controllo degli accessi.

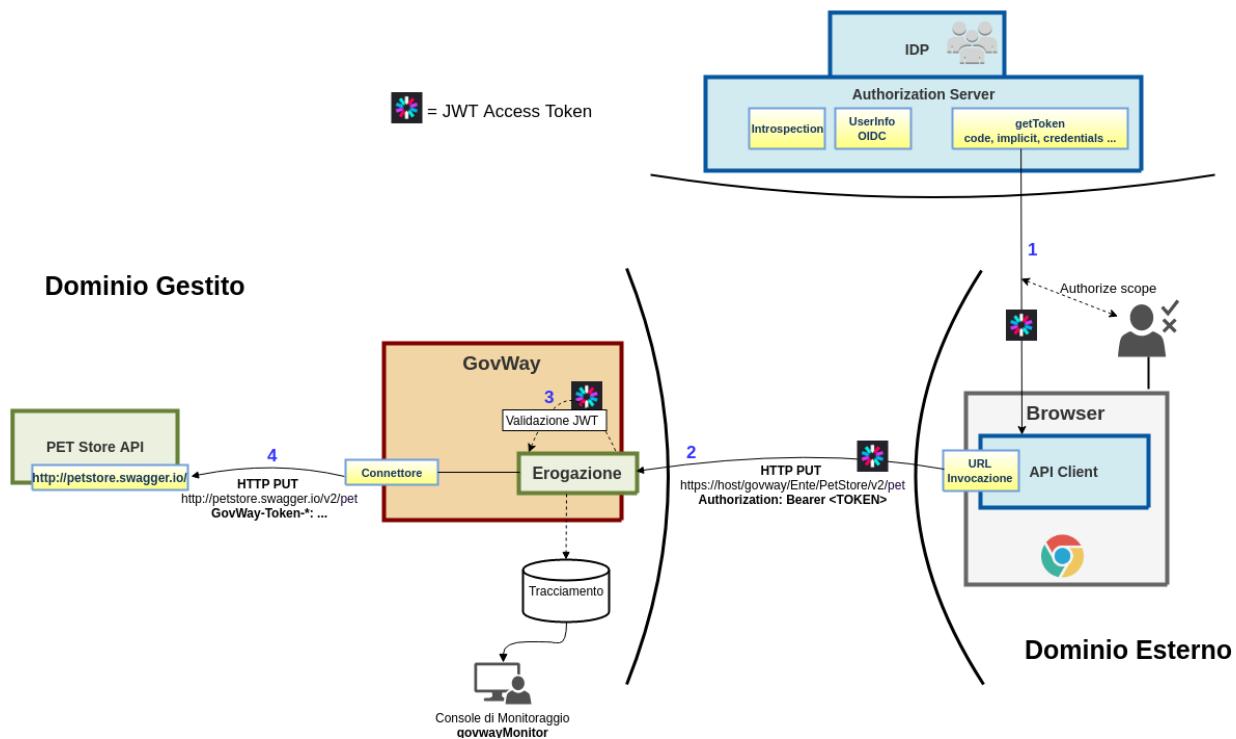


Fig. 2.7: Erogazione OAuth

4. Se la validazione è superata, Govway inoltra la richiesta al servizio erogatore.

2.2.1 Esecuzione

Facendo riferimento al progetto Postman è possibile verificare direttamente l'esecuzione dei passi di questo scenario. Passi da eseguire:

1. All'inizio possiamo verificare come il client non riesca ad accedere al servizio senza l'utilizzo del token. La request «IN token-auth (postPet) Error» effettua una chiamata alla risorsa «POST /pet» in assenza del token richiesto. Govway respinge la richiesta con la restituzione dell'errore mostrato in Fig. 2.8.
2. Successivamente si passa alla chiamata della «POST /pet» seguendo il flusso OAuth2 richiesto per l'approvvigionamento del token di autorizzazione. Posizionarsi sulla request «IN token-auth (postPet) OK»:
 - Nella sezione «Authorization» selezionare il Type «OAuth 2.0» e premere il pulsante «Get New Access Token»
 - La maschera fornita (Fig. 2.9) deve essere compilata con i parametri necessari ad richiedere un token all'authorization server. Utilizzare i seguenti parametri che permettono di richiedere un token all'authorization server preconfigurato per lo scenario:

```

Callback URL: {{keycloak-callback-url}}
Auth URL: {{keycloak-url-auth}}
Access Token URL: {{keycloak-url-token}}
Client ID: {{keycloak-client-id}}
Client Secret: {{keycloak-client-secret}}
  
```

- Compilati correttamente i campi per ottenere un token cliccare sul pulsante «Request Token»

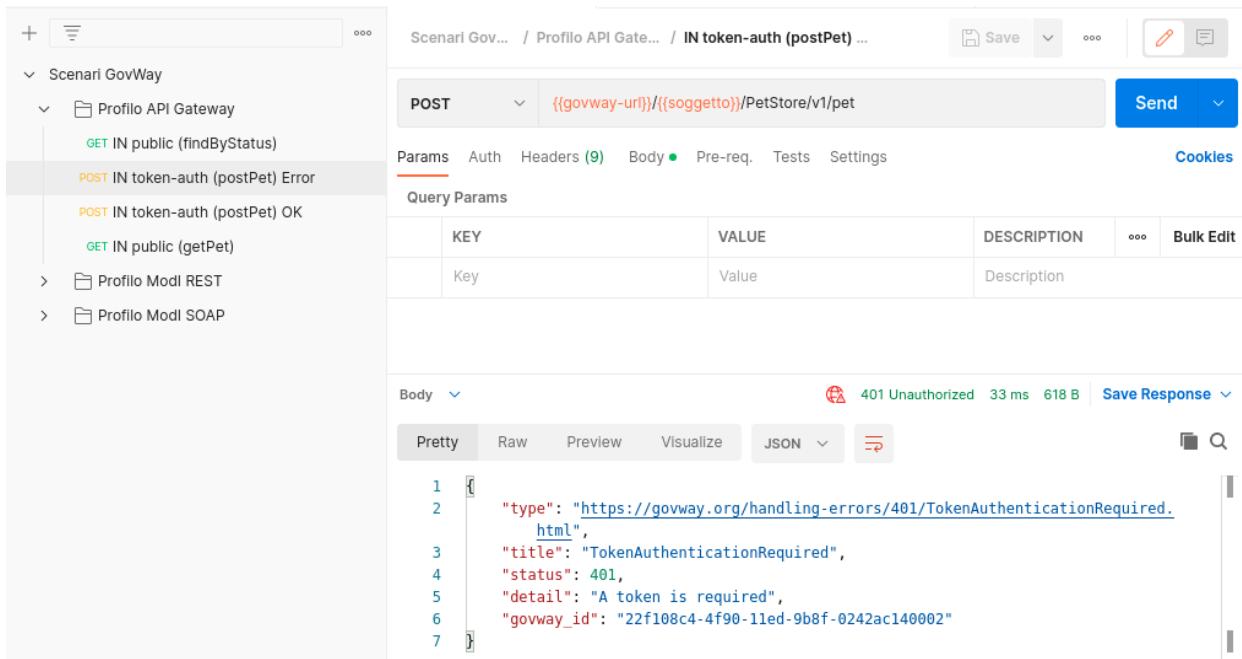


Fig. 2.8: Invocazione della POST /pet senza token

- Completare il processo di autenticazione dell’utente seguendo il flusso proposto ed utilizzando le credenziali dell’utente preconfigurato sull’authorization server per lo scenario di test:

```

username: paolorossi
password: 123456

```

- Superata l’autenticazione, viene restituito l’access token (mostrato a video sulla finestra popup).
- Inserire il token nella richiesta premendo il pulsante «Use Token».
- Eseguire la richiesta tramite il pulsante «Send».
- L’operazione viene eseguita con successo e restituito l’esito (Fig. 2.10).

3. Possiamo verificare che le limitazioni sull’accesso non sono efficaci nel caso di invocazione di operazioni di lettura. Il passo «IN public (getPet)» esegue una GET. Si noti come la sezione Authorization abbia l’impostazione del Type su «No Auth». Questa request legge il dato creato con la POST precedente e, come è possibile riscontrare al termine dell’esecuzione, viene correttamente eseguita in assenza di credenziali (Fig. 2.11).

2.2.2 Configurazione

L’erogazione è già stata preconfigurata per prevedere un controllo degli accessi differente tra le risorse che riguardano operazioni di scrittura (POST, PUT, DELETE) e le risorse che riguardano solo letture (GET).

Di seguito vengono descritti i passi che sono stati effettuati per arrivare alla configurazione esistente partendo dall’erogazione configurata con accesso pubblico.

I passi di configurazione finalizzati a limitare l’accesso alle sole operazioni di scrittura sono i seguenti:

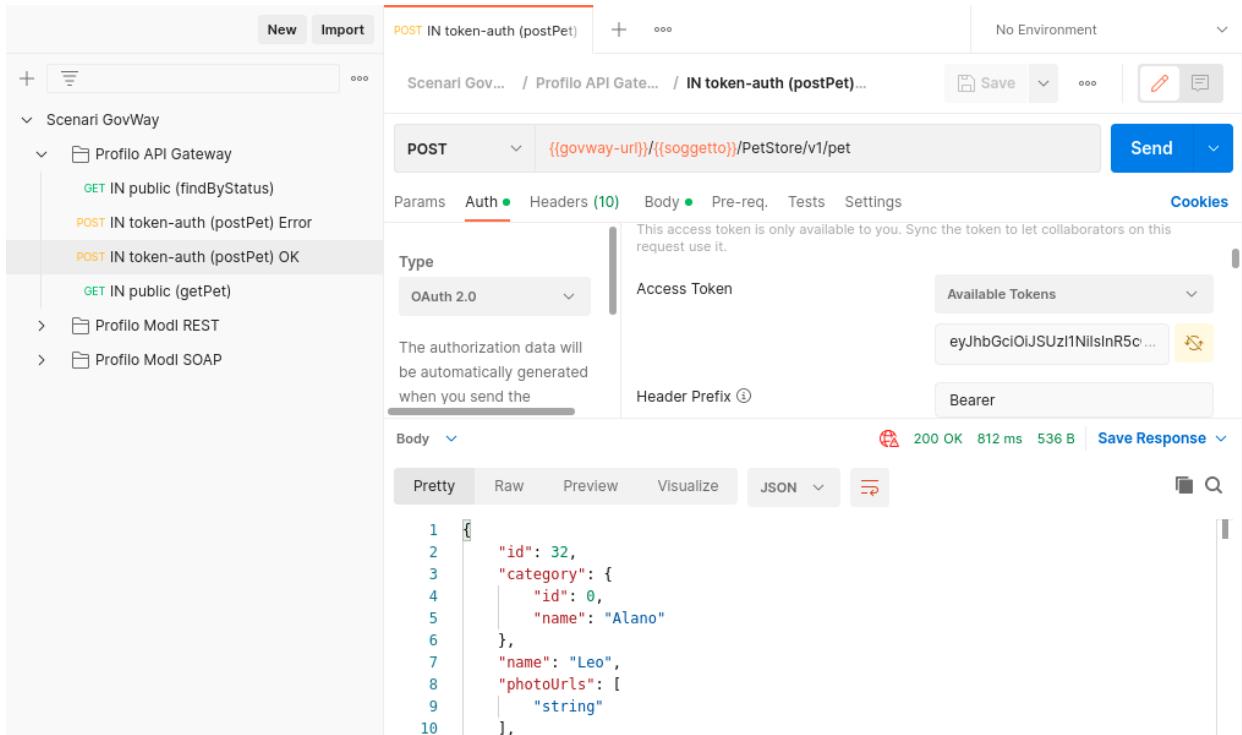
1. Dal dettaglio dell’erogazione, si procede con la creazione di una nuova configurazione, cui diamo il nome «Scritture» (Fig. 2.12).

GET NEW ACCESS TOKEN X

Token Name	<input type="text"/>
Grant Type	Authorization Code ▼
Callback URL i	<input type="text"/> {{keycloak-callback-url}}
Auth URL i	<input type="text"/> {{keycloak-url-auth}}
Access Token URL i	<input type="text"/> {{keycloak-url-token}}
Client ID i	<input type="text"/> {{keycloak-client-id}}
Client Secret i	<input type="text"/> {{keycloak-client-secret}}
Scope i	<input type="text"/> e.g. read:org
State i	<input type="text"/> State
Client Authentication	Send as Basic Auth header ▼

Request Token

Fig. 2.9: Ottenimento nuovo token



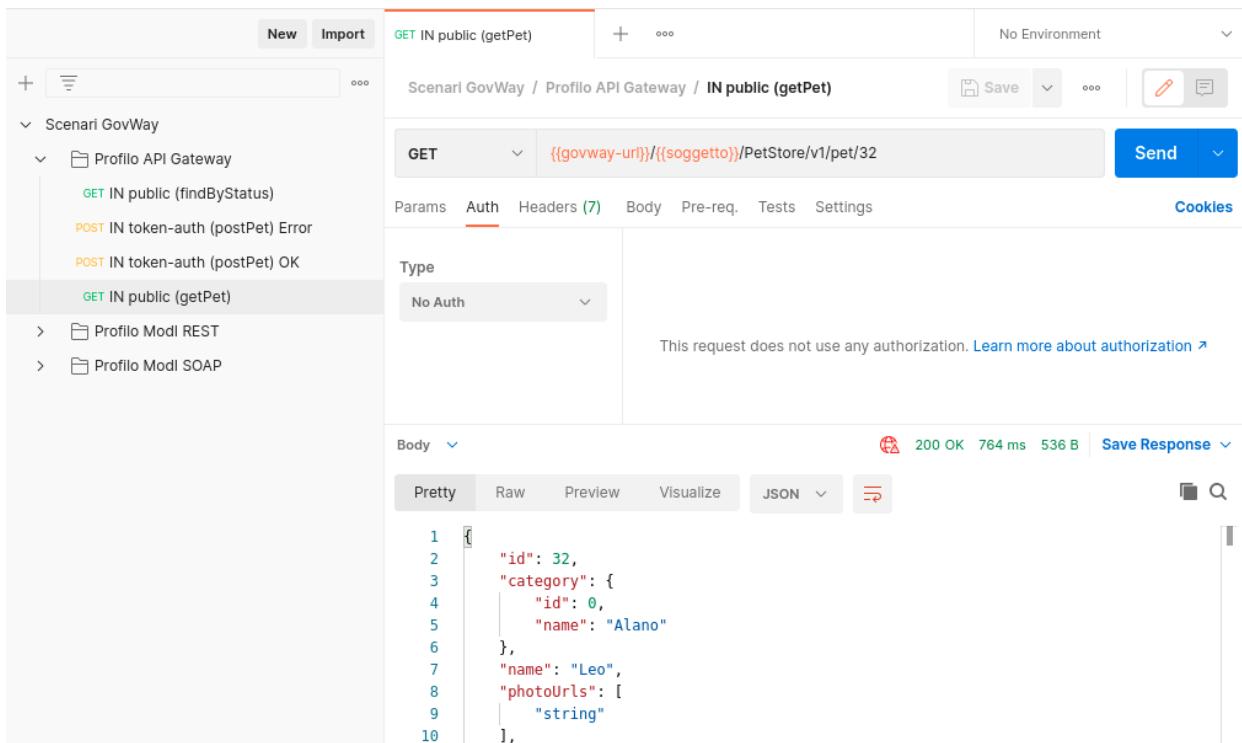
The screenshot shows the Scenari Applicativi interface for a POST request to the '/pet' endpoint. The request is labeled 'IN token-auth (postPet)'. The 'Auth' tab is selected, showing 'OAuth 2.0' as the type. The 'Access Token' field contains a token: 'eyJhbGciOiJSUzI1NiIsInR5c...'. The 'Header Prefix' is set to 'Bearer'. The response status is 200 OK with 812 ms and 536 B. The response body is a JSON object:

```

1  {
2    "id": 32,
3    "category": {
4      "id": 0,
5      "name": "Alano"
6    },
7    "name": "Leo",
8    "photoUrls": [
9      "string"
10 ]

```

Fig. 2.10: Invocazione della risorsa “POST /pet” con token



The screenshot shows the Scenari Applicativi interface for a GET request to the '/pet/32' endpoint. The request is labeled 'IN public (getPet)'. The 'Auth' tab is selected, showing 'No Auth'. The response status is 200 OK with 764 ms and 536 B. The response body is the same JSON object as in Fig. 2.10:

```

1  {
2    "id": 32,
3    "category": {
4      "id": 0,
5      "name": "Alano"
6    },
7    "name": "Leo",
8    "photoUrls": [
9      "string"
10 ]

```

Fig. 2.11: Invocazione della risorsa “GET /pet/id” con token

- Selezionare dall’elenco delle risorse quelle che riguardano operazioni di scrittura (POST, PUT, DELETE)
- Indicare per la *Modalità* il valore «*Nuova*» e quindi selezionare «*autenticato*» nel campo *Accesso API*

Erogazioni > PetStore v1 (Test) > Configurazione > Aggiungi

Configurazione

Note: (*) Campi obbligatori

Nome Gruppo * Scrittura

Risorse *

- POST /pet
- PUT /pet
- GET /pet/findByStatus
- GET /pet/findByTags
- DELETE /pet/{petId}
- GET /pet/{petId}
- POST /pet/{petId}
- POST /pet/{petId}/uploadImage
- GET /store/inventory
- POST /store/order

Modalità Nuova

Controllo degli Accessi

Accesso API autenticato

SALVA

Fig. 2.12: Creazione di una configurazione specifica per le operazioni di scrittura

2. Nella nuova configurazione «*Scrittura*» si va ad aggiornare la sezione «*Controllo Accessi*» effettuando le seguenti azioni (Fig. 2.13):
 - Abilitare l’autenticazione token selezionando la policy «*KeyCloak*» (configurazione preesistente per l’integrazione all’authorization server), lasciando invariate le altre opzioni del medesimo riquadro.
 - Disabilitare le altre funzionalità di controllo degli accessi: Autenticazione Trasporto, Autorizzazione e Autorizzazione Contenuti.
3. Dopo aver salvato la nuova configurazione, verificare il riepilogo delle informazioni, che devono corrispondere a quanto riportato in Fig. 2.14.

Erogazioni > PetStore v1 (Test) > Configurazione > Controllo Accessi del gruppo 'Scrittura'

Controllo Accessi del gruppo 'Scrittura'

Note: (*) Campi obbligatori

Autenticazione Token

Stato	abilitato
Policy *	KeyCloak
Token Opzionale	<input type="checkbox"/>
Validazione JWT	abilitato
Token Forward	abilitato

Required Claims

Issuer	<input type="checkbox"/>
ClientId	<input type="checkbox"/>
Subject	<input type="checkbox"/>
Username	<input type="checkbox"/>
eMail	<input type="checkbox"/>

Autenticazione Trasporto

Stato	disabilitato
-------	--------------

Autorizzazione

Stato	disabilitato
-------	--------------

Autorizzazione Contenuti

Stato	disabilitato
-------	--------------

SALVA

Fig. 2.13: Impostazione dell'autenticazione token nel controllo degli accessi

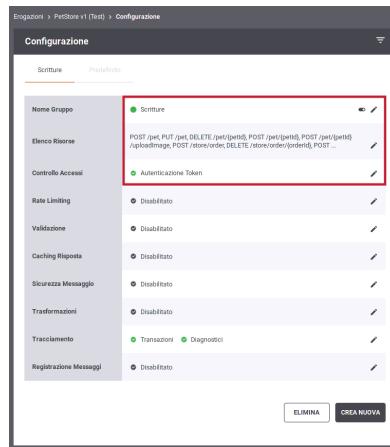


Fig. 2.14: Riepilogo della configurazione effettuata

CAPITOLO 3

Profilo “ModI”

Nelle sezioni successive verranno mostrati degli scenari di esempio di API Rest e API SOAP erogate o fruite con profilo “ModI” in accordo alla normativa prevista dal Modello di Interoperabilità.

I scenari descritti si differenziano rispetto ai pattern di sicurezza associati alle API erogate o fruite:

- nella sezione *Pattern “ID_AUTH”* le API sono configurate tramite il pattern modipa_idar01;
- nella sezione *Pattern “INTEGRITY”* viene utilizzato il pattern modipa_idar03;
- nella sezione *Pattern “ID_AUTH” via PDND* le API sono configurate tramite il pattern modipa_pdnd;
- infine nella sezione *Pattern “ID_AUTH” via PDND + “INTEGRITY”* viene utilizzato il pattern modipa_pdnd_integrity.

Nota: Per una consultazione mirata alle informazioni di interesse per lo scenario si consiglia di impostare nel menù in alto a destra il profilo “ModI” e la selezione del soggetto “Ente” come mostrato nella figura Fig. 2.1.



Fig. 3.1: Selezione del profilo “ModI”

3.1 Pattern “ID_AUTH”

Gli scenari riportati in questa sezione riguardano API configurate con pattern modipa_idar01.

3.1.1 Erogazione API REST

Obiettivo

Esporre un servizio, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza “ID_AUTH_REST_01” descritto nella sezione modipa_idar01.

Sintesi

Mostriamo in questa sezione come procedere per l'esposizione di un servizio REST da erogare nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede il trust del certificato X.509 in modo da assicurare sia a livello di canale che a livello di messaggio l'autenticazione e autorizzazione del fruitore.

La figura seguente descrive graficamente questo scenario.

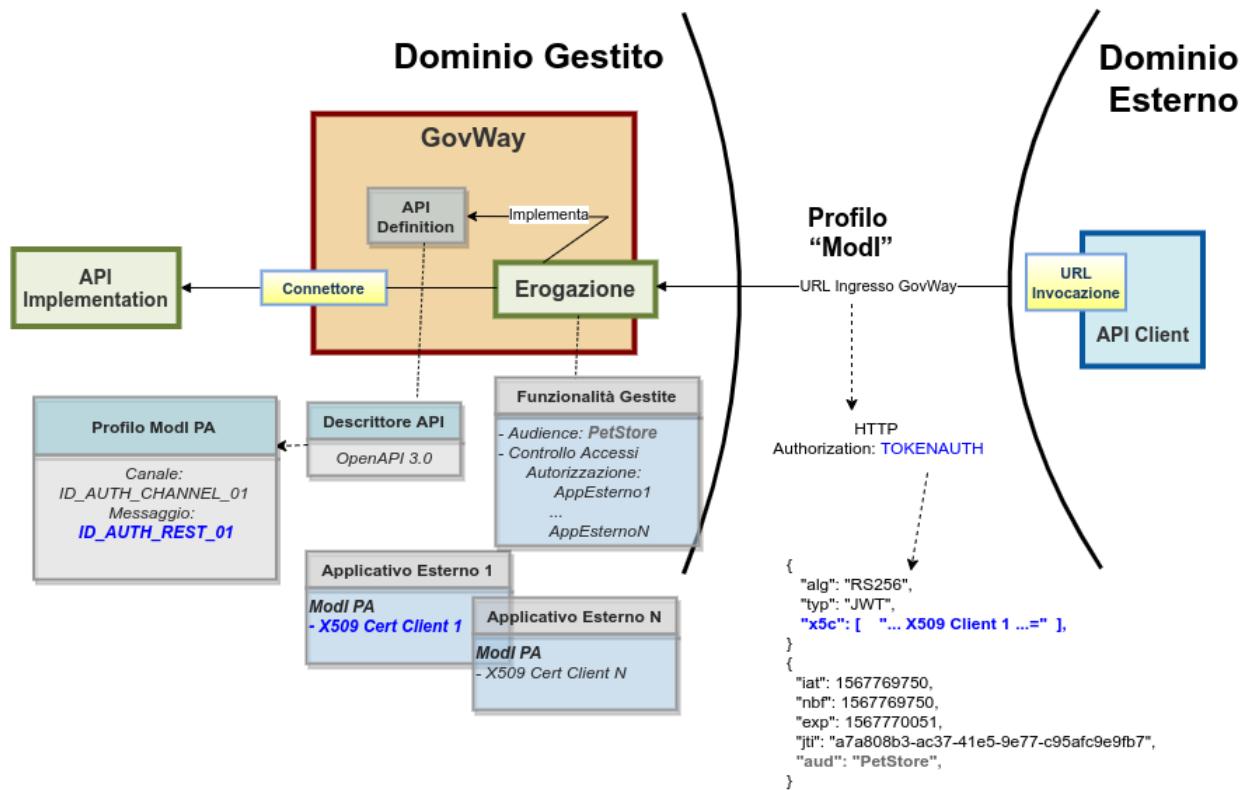


Fig. 3.2: Erogazione di una API REST con profilo “Modi”, pattern ID_AUTH_REST_01

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»

3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.3: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API REST di esempio (PetStore) definita con pattern di interazione “CRUD” e pattern di sicurezza «ID_AUTH_CHANNEL_02» e «ID_AUTH_REST_01»;
- un'istanza Govway per la gestione del profilo ModI nel dominio dell'erogatore;
- un client del dominio esterno che invoca la risorsa «POST /pet» dell'erogazione esposta da Govway;
- il server PetStore di esempio che riceve le richieste inoltrate dal Govway e produce le relative risposte. Per questo scenario viene utilizzato il server disponibile on line all'indirizzo “<https://petstore.swagger.io/>”.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

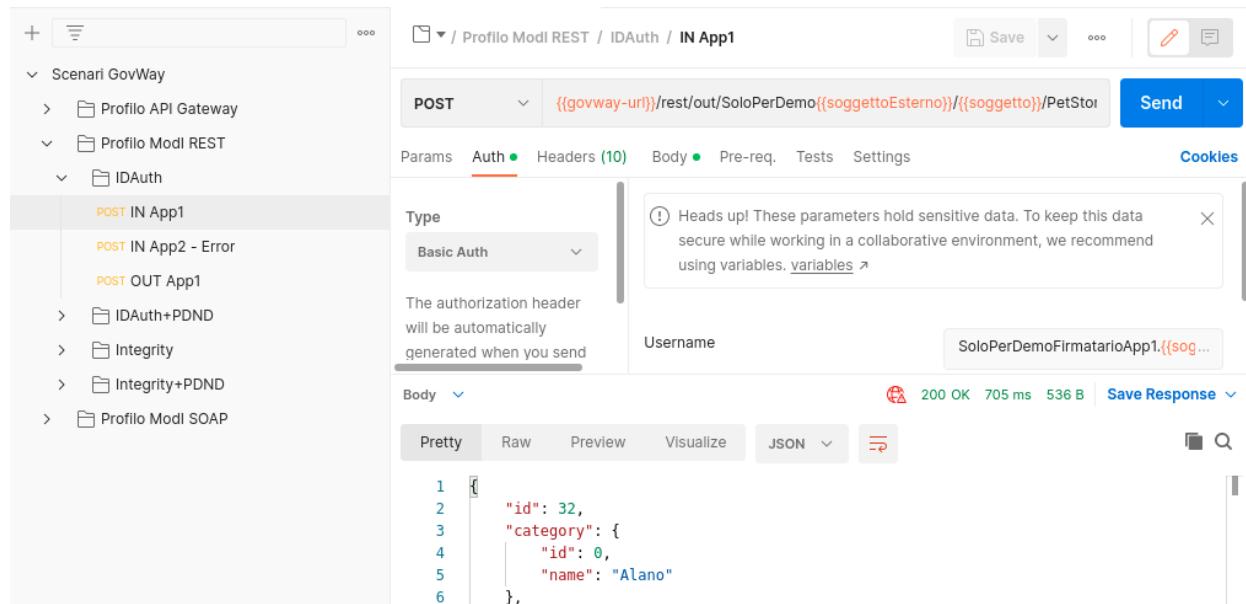


Fig. 3.4: Pattern IDAuth - Erogazione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

1. Lo scambio del messaggio con il dominio fruitore (comunicazione interdominio) avviene in accordo al pattern «ID_AUTH_CHANNEL_02» e quindi con protocollo SSL e autenticazione client. Dal dettaglio della transazione si possono consultare i messaggi diagnostici dove è visibile la fase di autenticazione del client con i dati di validazione del certificato ricevuto (Fig. 3.5).

2019-10-01 14:29:03.352	infoIntegration	RicezioneBuste	Ottenute credenziali di accesso (SSL-Subject 'CN=enteEsterno.govway.org, O=govway.org, C=it') fornite da Traefik
2019-10-01 14:29:03.352	infoIntegration	RicezioneBuste	Autenticazione [ssl] in corso (SSL-Subject 'CN=enteEsterno.govway.org, O=govway.org, C=it') ...
2019-10-01 14:29:03.359	infoIntegration	RicezioneBuste	Autenticazione [ssl] effettuata con successo

Fig. 3.5: Sicurezza canale «ID_AUTH_CHANNEL_02»

2. Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruitore, come in Fig. 3.6. Come si nota, al payload JSON è associato un insieme di header HTTP tra i quali «Authorization», che contiene il token di sicurezza.
3. Grazie alle configurazioni presenti nell'erogazione, ed in particolare alla relazione di trust stabilita con il fruitore, Govway è in grado di validare i dati di sicurezza ricevuti andando a decodificare il token. Nella fase di validazione del token si può notare come la sezione header (Fig. 3.7) riporti l'identità del fruitore e il suo certificato X.509, mentre la sezione payload (Fig. 3.8) contenga i riferimenti temporali (iat, nbf, exp) e l'audience (aud).
4. Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruitore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso. Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Le evidenze del processo di validazione sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.9). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dal token di sicurezza presente nel messaggio.
5. Dopo l'inoltro al servizio erogatore, Govway riceve la risposta e la elabora producendo il relativo token di sicurezza utilizzando le impostazioni di firma fornite nell'ambito dell'erogazione relativamente all'elaborazione della risposta. Sulla console govwayMonitor è possibile visualizzare il messaggio di risposta in uscita, dove si rileva la presenza del token prodotto nell'header HTTP «Authorization» (analogamente a Fig. 3.6).
6. Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo “App1-ModI” identificato grazie al certificato X.509 presente all'interno del token. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. La trasmissione è basata sul pattern «ID_AUTH_CHANNEL_02», riguardo la sicurezza canale, come evidenziato nei messaggi diagnostici dalla presenza degli elementi dell'handshake SSL e relativi dati dei certificati scambiati (Fig. 3.5).
2. La sicurezza messaggio applicata è quella del pattern «ID_AUTH_REST_01» come ampiamente mostrato nelle tracce dei messaggi di richiesta e risposta, dove sono presenti i certificati degli applicativi e le firme dei payload (e le relative validazioni).
3. L'identificazione del fruitore avviene rispetto al certificato X.509 presente all'interno del token. È stato anche mostrato come sia possibile configurare criteri autorizzativi puntuali.

Headers	
Nome	
Content-Type	application/json
X-Message-Id	1f46c4b4-4f9b-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	cde738cd-acfc-4785-a59a-eb751595a001
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybmc8uZ292d2F5Lm9y h2UWZIHrQDLuBSuHsJQWfc2Wp16rbtLxvMqKSONk6lxmWknBch1hXBwzeTmPAkNHcDoYpqhmdR
X-Forwarded-Port	443
Pragma	no-cache
Accept-Encoding	gzip, deflate, br

Fig. 3.6: Messaggio inviato dal fruttore

HEADER: ALGORITHM & TOKEN TYPE

```

ID  {
  "alg": "RS256",
  "typ": "JWT",
  "kid": "app1.enteesterno.govway.org",
  "x5c": [
    "MIIE/jCCAuagAwIBAgICAPgwDQYJKoZIhvcNAQELBQAwNjELMAkGA1
    UEBhMCAxQxEzARBgNVBAoMCmdvdndheS5vcmcxEjAQBgNVBAMMCUdvd
    1dheSBDQTAeFw0yMjEwMTkwNzU1NThaFw0zNzEwMTUwNzU1NThaMEgx
    CzAJBgNVBAYTAm10MRMwEQYDVQQDApnb3Z3YXkub3JnMSQwIgYDVQQ
    DDBthcHAxLmVudGVFc3R1cm5vLmdvdndheS5vcmcwggEiMA0GCSqGSI
    b3DQEBAQUAA4IBDwAwggEKAoIBAQC1/cfENX06hdvEVxJiJAF00ePjn
    5Sh/HIJ2du8hRv0zA+KFFieaF4xh1mS0T1oq/vwdxFqvcd2k1bTJ37r
    jBo6DKuQZor83j/Do87x3sFJe/epGKx96Q3PRE9mA1qx3Y5FFShfGNZ
    zG2RFNA2jhVQ/bs8d9E051FC3XshF90CtJJs9LGvT2+0+uJK3siA6ht
    KcYQ58UcK1W1Y109MnXqaz82TiH93eTSkk33w0A9atzC0w3JAVmcRRk
    d0hFBjMeEvNR86cdNfy9Xit7ZDR11IB8tel0/fI/oAW0oK/3TbF1XOr
    VL1QhMc1JdqS3NwJLAyoqmZT/Xh5DqjDi7ldghwbAgMBAAGjggECMIH
    /MAkGA1UdEwQCMAAwEQYJYIZIAYb4QgEBBAQDAgeAMDMGCWCGSAGG+E
    IBDQQmFiRPcGVuU1NMIEd1bmVyYXR1ZCBDbG11bnQgQ2VydG1maWNhd
    GUwHQYDVR0OBYEFCBwk8Bs9JS+6c/vTU+JX0eqX81BMGYGA1UdIwRf
    MF2AFCqHFNpm2RdIA3igRXzNEeJ5ivegoTqk0DA2MQswCQYDVQQGEwJ
    pdDETMBEGA1UECgwKZ292d2F5Lm9yZzESMBAGA1UEAwJR292V2F5IE
    NBggkA4tGAdmeSJF4wDgYDVR0PAQH/BAQDAgXgMBMGA1UdJQQMMAoGC
    CsGAQUFBwMCMA0GCSqGSIb3DQEBCwUAA4ICAQDRj52cdYwcqFDNmC29
    CY0DR0N0TM/5RKq9sL6sgI7z4cUmkyIeGh/9YQDoRFhDBVGZ80rx0ka
    sZ/Po0Iuw+41f9IDTBe04Ym0CK3M1M9H2LiEKe9hngRtjzGw5tFRQnq
    KbLLX61otJAXuE488SrSAMbEdez1bZt+V1Sgc48f0KsjShUs8CwSW0G
    6RE5w4Q4oa0dX971PTziWDoFnxBfN17/HAYA0625/vcp8PrZLqhTIGH
    7dt+1T4Hb+i10wKBS7B8Cab0Gh0spiHDDGNEYX50d1ZYmWJQ10ysK6
    1Yx1WtCrKPfmsvSeqiVxJPHUgwTsFPrgoVRt+dT1NnAdXYxFk0Yxz7z
    n7qeKDi6cXHLTsYet1cQfedyDPE0rli4GFL1KY37NFqRtJx5NadkJk6
    GXk43zIFQo119PGJ8nVHupB6IBJ1h/6xem1TTMSt52zcjV4b5zRHL8Z
    qNF+S0QnJKcH2FcyAYuGjuVj0qa5rhi5wNcy7ilcDShM8tsPJ5qpW1M
    E0mhmWVY+w5KBCpMoLBn9cvqAn/N19L3e3SqH1Klp8Hw05CtH4/tLEe
    3N+0z+8xzcmLdqbaZ9nD7YVLVoyt5Y+Ixuj17F18dzEh9dzLhJojsB
    mPjoFMMMyulbpjZG0A1TjKVpkxyXgaqsd9Hjs4ATg79Vk8U/GnEXJhXQ
    xU2TYw=="
  ],
  "x5t#S256": "agRQxqs-
  VYDP2NIzbR7XH2GiInWH2bcL1xMPhimfMKk"
}

```

Fig. 3.7: Sezione «Header» del Token di sicurezza

```

PAYLOAD: DATA

{
  "iat": 1666176318,
  "nbf": 1666176318,
  "exp": 1666176378,
  "jti": "1f46c4b4-4f9b-11ed-a5ac-0242ac140002",
  "aud": "petstore.ente.govway.org",
  "client_id": "app1.enteesterno.govway.org",
  "iss": "SoloPerDemoEnteEsterno",
  "sub": "SoloPerDemoFirmatarioApp1"
}

```

Fig. 3.8: Sezione «Payload» del Token di sicurezza

Informazioni Modelli

Sicurezza Messaggio ID_AUTH_REST_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Accesso CRUD

Sicurezza Messaggio

ClientId app1.enteesterno.govway.org

Subject SoloPerDemoFirmatarioApp1

Issuer SoloPerDemoEnteEsterno

MessageId 1f46c4b4-4f9b-11ed-a5ac-0242ac140002

Audience petstore.ente.govway.org

NotBefore 2022-10-19_12:45:18.000

Expiration 2022-10-19_12:46:18.000

IssuedAt 2022-10-19_12:45:18.000

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Fig. 3.9: Traccia della richiesta elaborata dall'erogatore

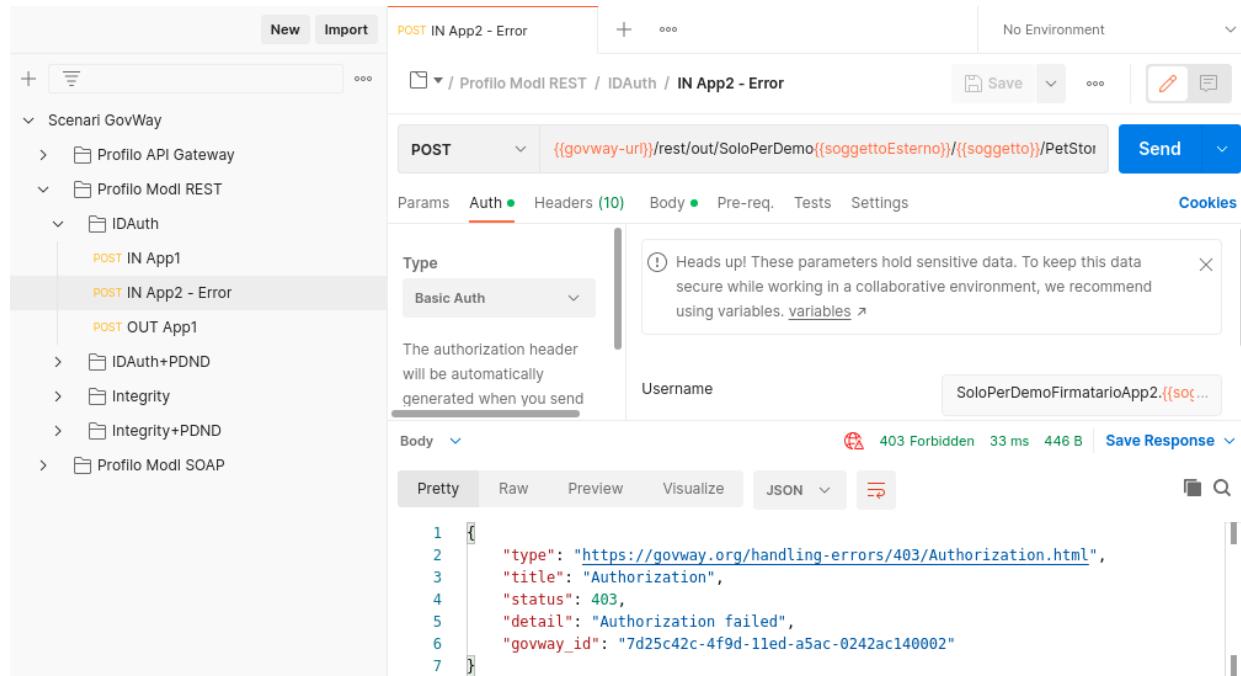


Fig. 3.10: Pattern IDAuth - Erogazione API REST - Autorizzazione negata, esecuzione da Postman

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.11: Profilo ModI della govwayConsole

Registrazione API

Viene registrata l’API «PetStoreAuth» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.12).

Applicativo Esterno

È opzionalmente possibile registrare l’applicativo esterno che corrisponde al fruitore del servizio. Questa scelta può essere fatta in base al tipo di autorizzazione che si è impostata sui fruitori. Vediamo i seguenti casi:

- Se il truststore utilizzato da Govway per l’autenticazione dei fruitori (sicurezza messaggio) contiene i singoli certificati degli applicativi autorizzati, questo passo può anche essere omesso. La gestione del truststore è sufficiente a stabilire i singoli fruitori autorizzati.

API > PetStoreAuth v1 > Profilo Interoperabilità

Profilo Interoperabilità

Note: (*) Campi obbligatori

Modi

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_02

Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern: ID_AUTH_REST_01

Direct Trust con certificato X.509

Header HTTP del Token: Authorization Bearer

Applicabilità: Richiesta e Risposta

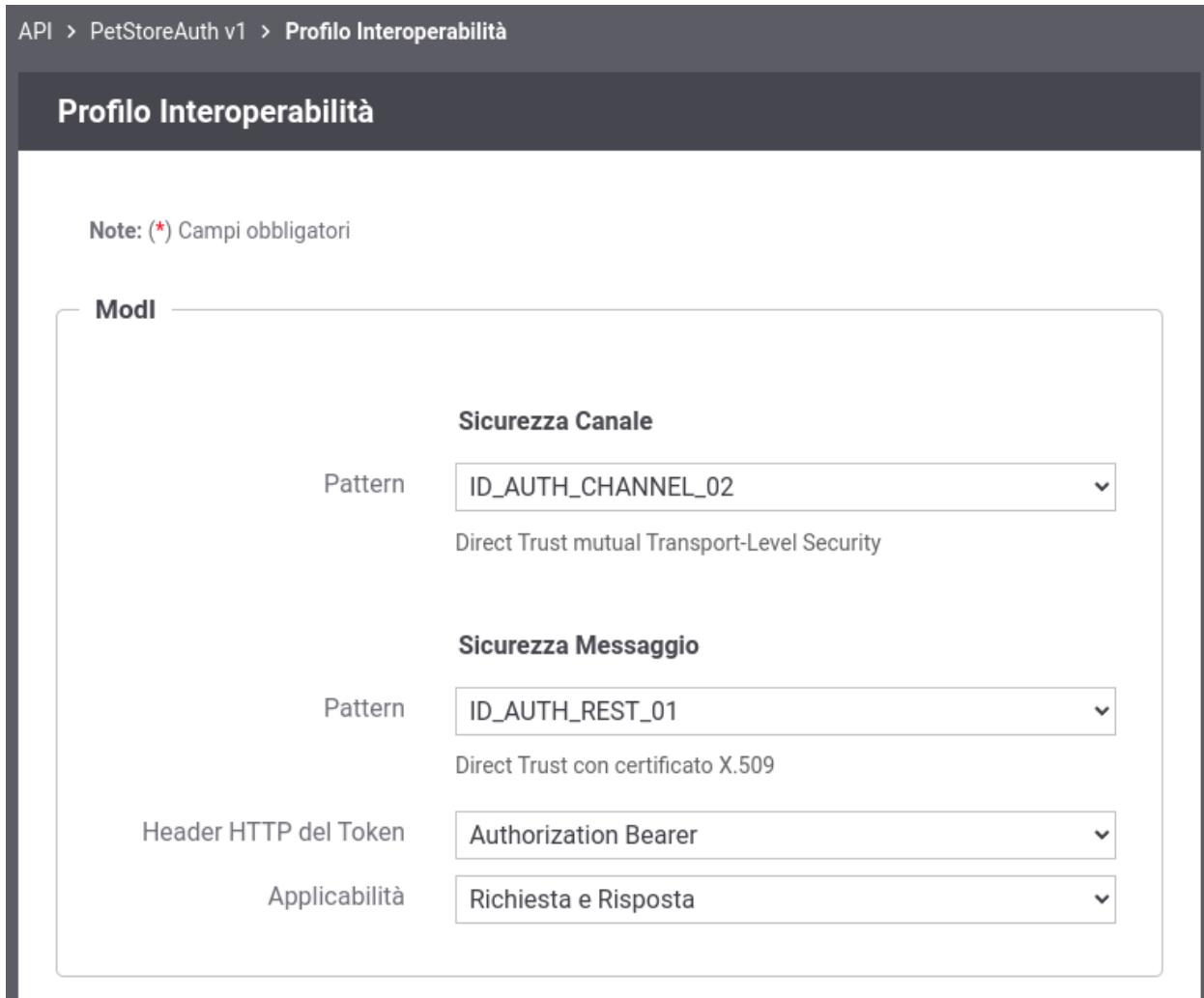


Fig. 3.12: Configurazione Pattern ModI «ID_AUTH_REST_01» sulla API REST

- Se il truststore contiene la CA emittente dei certificati utilizzati dai fruitori, l'autorizzazione puntuale non è possibile a meno di non procedere con la registrazione puntuale degli applicativi fornendo i singoli certificati necessari per l'identificazione (Fig. 3.13). Questo scenario è quello preconfigurato.

Erogazione

Si registra l'erogazione «PetStoreAuth», relativa all'API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.14). In questo contesto vengono inseriti i dati necessari per validare le richieste in ingresso.

La sezione «ModI Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza da inserire nel messaggio di risposta (Fig. 3.15).

Se si è scelto di registrare gli applicativi esterni, fruitori del servizio, è possibile intervenire sulla configurazione del «Controllo degli Accessi» per l'erogazione, in modo da specificare i singoli applicativi fruitori autorizzati ad effettuare richieste al servizio erogato. Questo scenario è quello preconfigurato come mostrato nelle figure Fig. 3.16 e Fig. 3.17.

3.1.2 Fruizione API REST

Obiettivo

Fruire di un servizio REST, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza “ID_AUTH_REST_01” descritto nella sezione modipa_idar01.

Sintesi

Mostriamo in questa sezione come procedere per l'integrazione di un applicativo con un servizio REST erogato nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede il trust del certificato X.509 in modo da assicurare sia a livello di canale che a livello di messaggio l'autenticazione e autorizzazione del fruitore.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.19: Profilo ModI della govwayMonitor

Applicativi > App1-Modl

App1-Modl

Note: (*) Campi obbligatori

Applicativo

Dominio	Esterno
Soggetto	EnteEsterno
Nome *	App1-Modl
Tipo	Client
<u>Proprietà(0)</u>	

Ruoli

[visualizza\(0\)](#)

Modl

Sicurezza Messaggio	Authorization Modl
Certificato	
Cambia Certificato	
Aggiungi Certificato	
Download	
Verifica	<input checked="" type="checkbox"/>
Subject	/c=it/cn=app1.enteEsterno.govway.org/o=govway.org/
Issuer	/c=it/cn=GovWay CA/o=govway.org/
Serial Number	248 (Hex) 00:F8
Self Signed	No
Not Before	19/10/2022 09:55:00
Not After	15/10/2037 09:55:00

Fig. 3.13: Configurazione applicativo esterno (fruitore)

Modi PA - Richiesta

Profilo Sicurezza Messaggio

Riferimento X.509	x5c (Certificate Chain) x5t#256 (Certificate SHA-256 Thumbprint) x5u (URL)
TrustStore Certificati	Default
Audience	PetStore

Se non viene fornito un valore, il valore atteso all'interno del security token corrisponderà all'url di invocazione

Fig. 3.14: Configurazione richiesta dell'erogazione

Modi PA - Risposta

Profilo Sicurezza Messaggio

Algoritmo	RS256
HTTP Headers da firmare *	Digest <input checked="" type="checkbox"/> Content-Type <input checked="" type="checkbox"/> Content-Encoding <input checked="" type="checkbox"/>

Riferimento X.509

Utilizza impostazioni della Richiesta

KeyStore

Default

Time to Live (secondi) *

300

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Fig. 3.15: Configurazione risposta dell'erogazione

Erogazioni > PetStoreAuth v1 (Ente) > Configurazione > Controllo Accessi

Controllo Accessi

— ▾ Autenticazione Token —

^ Autenticazione Canale

Stato https

— ▾ Autorizzazione —

Stato abilitato

Autorizzazione Canale

per Richiedente

Soggetti (1)

per Ruoli

Autorizzazione Messaggio

per Richiedente

Applicativi (1)

per Ruoli

Fig. 3.16: Controllo accessi con autorizzazione degli applicativi esterni

Erogazioni > PetStoreAuth v1 (Ente) > Configurazione > Controllo Accessi > Autorizzazione Messaggio - Applicativi

Autorizzazione Messaggio - Applicativi

Visualizzati record [1-1] su 1

	Soggetto	Applicativo	
<input type="checkbox"/>	EnteEsterno	App1-Mod1	<input type="checkbox"/>

Fig. 3.17: Lista degli applicativi esterni autorizzati

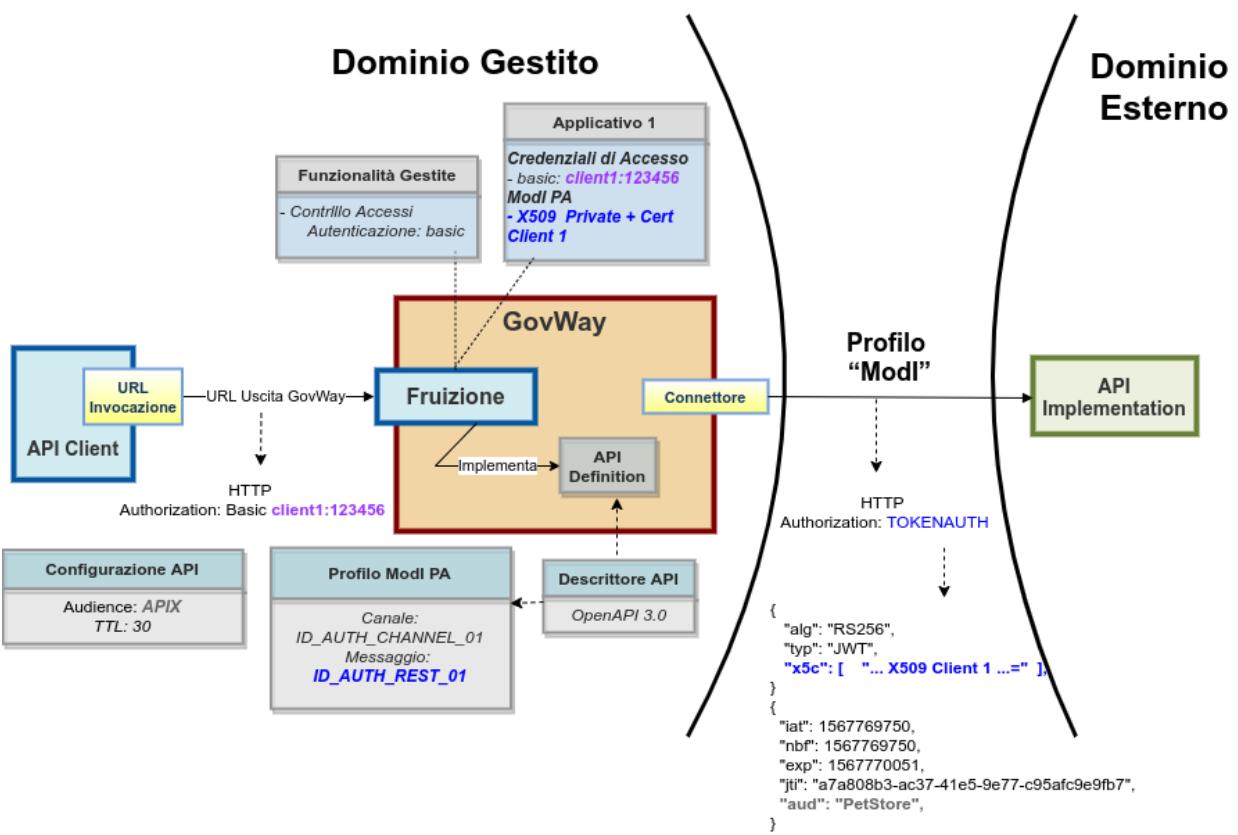


Fig. 3.18: Fruizione di una API REST con profilo “ModI”, pattern ID_AUTH_REST_01

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API REST di esempio (PetStore) definita con pattern di interazione “CRUD” e pattern di sicurezza «ID_AUTH_CHANNEL_02» e «ID_AUTH_REST_01»;
- un'istanza Govway per la gestione del profilo ModI nel dominio del fruitore;
- un client che invoca la risorsa «POST /pet» con un messaggio di esempio diretto alla fruizione configurata su Govway.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Fig. 3.20: Pattern IDAuth - Fruizione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto nelle diverse fasi dell'esecuzione andando a consultare la console “govwayMonitor”.

1. Il messaggio di richiesta inviato dal fruitore viene elaborato da Govway che, tramite la configurazione della firma digitale associata all'applicativo mittente, è in grado di produrre il token di sicurezza da inviare con la richiesta all'erogatore. Da govwayMonitor si può visualizzare il messaggio di richiesta in uscita che è il medesimo di quello in entrata con la differenza che è stato aggiunto il token di sicurezza tra gli header HTTP (Fig. 3.21).
2. L'header e il payload del token sono identici a quelli visualizzati nello scenario di erogazione REST, relativamente al messaggio in ingresso (Fig. 3.7 e Fig. 3.8). Le informazioni inserite nel token vengono anche tracciate e sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.22). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dal token di sicurezza presente nel messaggio.
3. Lo scambio del messaggio con il dominio erogatore (comunicazione interdominio) avviene in accordo al pattern «ID_AUTH_CHANNEL_02» e quindi con protocollo SSL e autenticazione client. Dal dettaglio della transazione si possono consultare i messaggi diagnostici dove è visibile la fase di apertura della connessione SSL (Fig. 3.23).

Headers	
Nome	
Content-Type	application/json
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
X-Forwarded-Port	443
Accept-Encoding	gzip, deflate, br
Postman-Token	d924391e-10cd-4c75-8063-4cbfaa74639a
User-Agent	GovWay
Accept	/*
GovWay-Message-ID	5ade2322-4fac-11ed-a5ac-0242ac140002
GovWay-Transaction-ID	5acd8134-4fac-11ed-a5ac-0242ac140002
Authorization	Bearer eyJhbGciOiJSUzI1NilsInR5cCl6lkpXVClsImtpZCI6ImFwcDEuZW50ZS5nb3Z3YXkub3JnWylSJxWAFBE4zpeb4JpJRwmafmwJLqddHy7j8bMjGx9x3lGOws6AhiTAKaK2HPGbpD

Fig. 3.21: Messaggio di richiesta in uscita (con token di sicurezza inserito nell'header HTTP)

Informazioni Modl	
Sicurezza Messaggio	ID_AUTH_REST_01
Sicurezza Canale	ID_AUTH_CHANNEL_02
Interazione	Accesso CRUD
Sicurezza Messaggio	
X509-Issuer	CN=GovWay CA, O=govway.org, C=it
X509-Subject	CN=app1.ente.govway.org, O=govway.org, C=it
Subject	App1-Modl
Issuer	Ente
ClientId	app1.ente.govway.org
Audience	petstore.enteEsterno.govway.org
MessageId	5ade2322-4fac-11ed-a5ac-0242ac140002
Expiration	2022-10-19_14:49:39.000
NotBefore	2022-10-19_14:48:39.000
IssuedAt	2022-10-19_14:48:39.000

Fig. 3.22: Traccia della richiesta generata dal fruitore

2019-09-16 16:36:11.209	infoProtocol	InoltroBuste	Invio Messaggio di cooperazione con identificativo [f26754d8-d596-476b-bc5b-5c1b2b95966b] in corso (location: https://auth03.govcloud.it/govway/rest/EnteEsterno/PetStore/v1/pet http-method:POST) ...
----------------------------	--------------	--------------	---

Fig. 3.23: Sicurezza canale «ID_AUTH_CHANNEL_02» sulla fruizione

4. Govway riceve la risposta dell'erogatore, dalla quale estrae il token di sicurezza al fine di effettuare i relativi controlli di validità e conservare la traccia. Consultando la traccia relativa alla trasmissione della risposta, sono visibili tra le altre informazioni i dati di autenticazione dell'erogatore e i riferimenti temporali.

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. La trasmissione è basata sul pattern «ID_AUTH_CHANNEL_02», riguardo la sicurezza canale, come evidenziato nei messaggi diagnostici dalla presenza degli elementi dell'handshake SSL e relativi dati dei certificati scambiati ([Fig. 3.23](#)).
2. La sicurezza messaggio applicata è quella dei pattern «ID_AUTH_REST_01», come ampiamente mostrato nelle tracce dei messaggi di richiesta e risposta, dove sono presenti i certificati degli applicativi e le firme dei payload (e le relative validazioni).

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.24: Profilo ModI della govwayConsole

Registrazione API

Viene registrata l'API «PetStoreAuth» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» ([Fig. 3.25](#)).

Applicativo

Si configura l'applicativo mittente indicando, nella sezione ModI, i parametri del keystore necessari affinché Govway possa produrre il token di sicurezza firmando per conto dell'applicativo ([Fig. 3.26](#)). Alla registrazione dell'applicativo vengono associate credenziali “basic” che consentono a GovWay di identificarlo.

Fruizione

Si registra la fruizione «PetStoreAuth», relativa all'API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» ([Fig. 3.27](#)). In particolare è possibile specificare l'audience atteso dall'erogatore e il tempo di validità del token.

La sezione «ModI Risposta» definisce i criteri per la validazione dei messaggi di risposta, come la posizione del token di sicurezza e il truststore per l'autenticazione dell'erogatore ([Fig. 3.28](#)).

API > PetStoreAuth v1 > Profilo Interoperabilità

Profilo Interoperabilità

Note: (*) Campi obbligatori

Modi

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_02

Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern: ID_AUTH_REST_01

Direct Trust con certificato X.509

Header HTTP del Token: Authorization Bearer

Applicabilità: Richiesta e Risposta

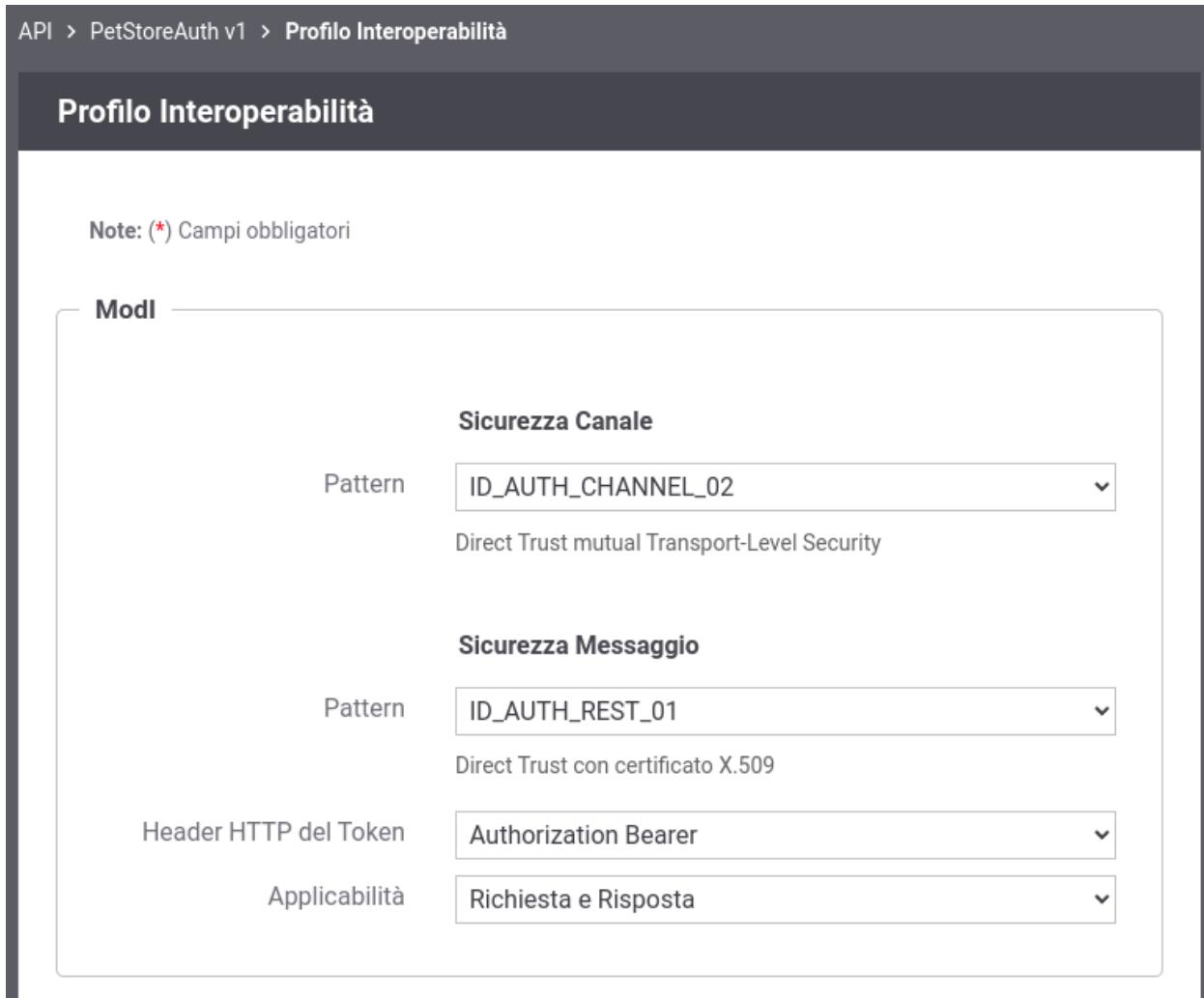


Fig. 3.25: Configurazione Pattern ModI «ID_AUTH_REST_01» sulla API

Applicativi > App1-Mod1

App1-Mod1

Note: (*) Campi obbligatori

Applicativo

Dominio	Interno
Soggetto	Ente
Nome *	App1-Mod1
Tipo	Client
Proprietà(0)	

Modalità di Accesso

Tipo	http-basic
Utente *	App1-Mod1.Ente
Modifica Password	<input type="checkbox"/>

Ruoli

[visualizza\(0\)](#)

Modi - Sicurezza Messaggio

KeyStore

Abilitato	<input checked="" type="checkbox"/>
Modalità	File System
Path *	/etc/govway/keys/keystore_app1.ente.pkcs12
Tipo	PKCS12
Password *	123456
Alias Chiave Privata *	app1.ente.govway.org
Password Chiave Privata *	123456

[Certificato](#)

Authorization Mod1

Identificativo Client	app1.ente.govway.org	i
-----------------------	----------------------	-------------------

Fig. 3.26: Configurazione applicativo fruitore

Modi - Richiesta

Sicurezza Messaggio

Algoritmo	<input type="text" value="RS256"/>
Riferimento X.509	<input type="text" value="x5c (Certificate)"/> x5t#256 (Certificate SHA-256 Thumbprint) x5u (URL)
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	<input type="text" value="60"/>
Indica la validità temporale, in secondi, a partire dalla data di creazione del security token	
Audience	<input type="text" value="petstore.enteEsterno.govway.org"/> 
Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore	
Claims	<input type="text"/>
Indicare per riga i claims (nome=valore); visualizzare 'info' per maggiori dettagli 	

Fig. 3.27: Configurazione richiesta della fruizione

Modi - Risposta

Sicurezza Messaggio

Riferimento X.509	<input type="text" value="Utilizza impostazioni della Richiesta"/>
TrustStore Certificati	<input type="text" value="Default"/>
Time to Live	<input type="text" value="Default"/>
Verifica Audience	<input checked="" type="checkbox"/> La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente
<input type="text"/> 	

Fig. 3.28: Configurazione risposta della fruizione

3.1.3 Erogazione API SOAP

Obiettivo

Esportare un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza "ID_AUTH_SOAP_01" descritto nella sezione modipa_idar01.

Sintesi

Mostriamo in questa sezione come procedere per l'esposizione di un servizio SOAP da erogare nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede il trust del certificato X.509 in modo da assicurare sia a livello di canale che a livello di messaggio l'autenticazione e autorizzazione del fruitore.

La figura seguente descrive graficamente questo scenario.

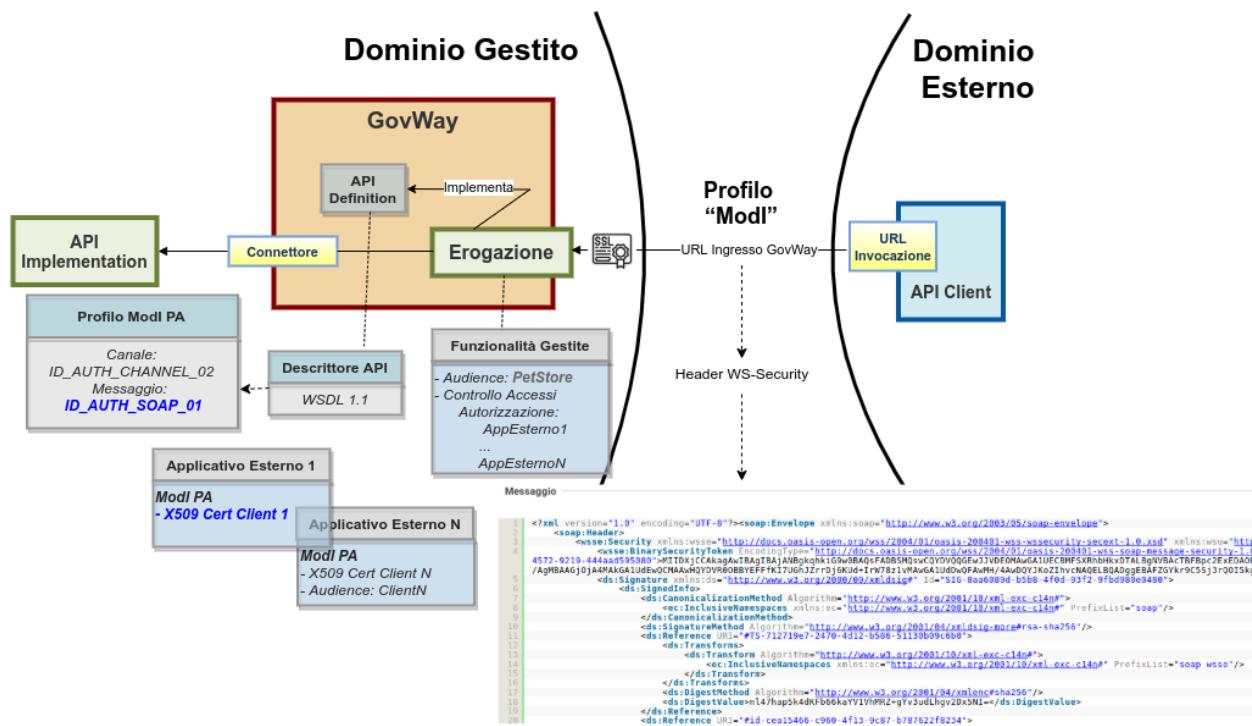


Fig. 3.29: Erogazione di una API SOAP con profilo “ModI”, pattern ID_AUTH_SOAP_01

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID
 2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
 3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_SOAP_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.30: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API SOAP di esempio (Credit Card Verification) definita con pattern di interazione Bloccante e pattern di sicurezza «ID_AUTH_CHANNEL_02» e «ID_AUTH_SOAP_01»;
- un'istanza Govway per la gestione del profilo ModI nel dominio dell'erogatore;
- un client del dominio esterno che invoca l'azione di esempio «CheckCC» dell'erogazione esposta da Govway;
- il server “Credit Card Verification” di esempio che riceve le richieste inoltrate dal Govway e produce le relative risposte. Per questo scenario viene utilizzato il server disponibile on line all'indirizzo “<https://ws.cdyne.com/creditcardverify/luhnchecker.asmx>”.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console govwayMonitor.

1. Per verificare l'utilizzo del canale SSL, in accordo al pattern «ID_AUTH_CHANNEL_02», si procede come già illustrato per [Esecuzione](#)
2. Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruitore, come in [Fig. 3.32](#). Come si nota, il messaggio SOAP contiene nell'header WS-Security sia il token di sicurezza (elemento «BinarySecurityToken») sia l'elemento «WSAddressing - To» prodotti dal fruitore con la relativa firma digitale (elemento «SignatureValue»).
3. Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruitore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso. Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Le evidenze del processo di validazione sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta ([Fig. 3.33](#)). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dal token di sicurezza presente nell'header soap.
4. Dopo l'inoltro al servizio erogatore, Govway riceve la risposta e la elabora producendo il relativo header ws-security da inserire nel messaggio di risposta. Sulla console govwayMonitor è possibile visualizzare il messaggio di risposta in uscita (analogamente a [Fig. 3.32](#)).
5. Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo “App1-ModI” identificato grazie al certificato X.509 presente all'interno dell'header WSSecurity. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

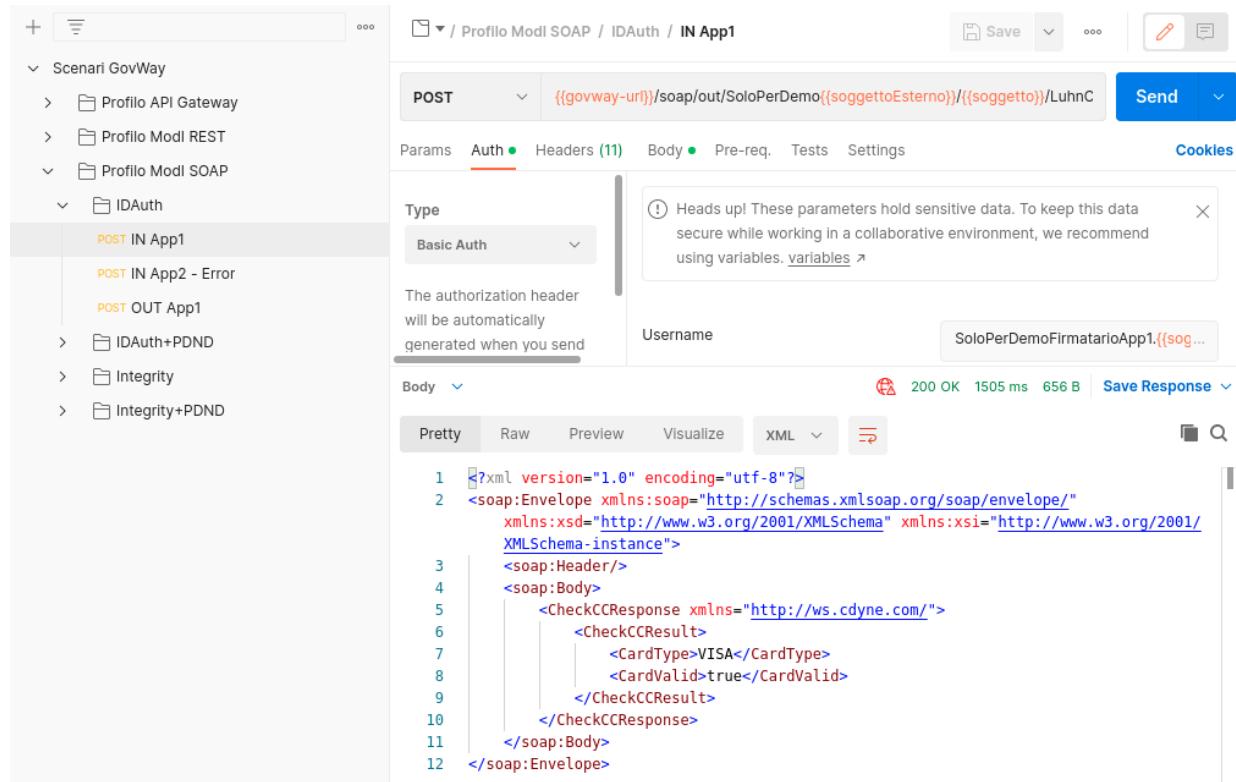


Fig. 3.31: Pattern IDAuth - Erogazione API SOAP, esecuzione da Postman

Messaggio

```

1  <?xml version="1.0" encoding="UTF-8"?><soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
2    <soap:Header>
3      <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#"
4        <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#"
c7761d94d64f">MIIE/zCAuegAwIBAgICAN4wDQYJKoZIhvNAQELBQAwNjELMAKGA1UEBMMCaXQxEARBgNVBAoCImdvndhe5VcmcxEjA0BgNVBAMMCUdvd1hneSB0DQTAf
/Wu06/YXIV1DHLYMjypb/fL0SL8SKa6uW9swPxcoGJP9aqw01v0/Bw2lpv1657h+BtNleBfhsmUnN17C25hBa/WlVkh78213f5LYC4s8H9nFc/faQQuuadLtxWohKwzNl
/ZaJBgNVHrMeAjAAMBEGCWCGSAGG+EIBAQEEAwIHqDAzBg1ghkgBvhvCA00EJhYKT38lbLNTTCBHZw5lcmF0ZWq02xpZw50IENlcnRpZmljYXRlMB0GA1UdDgQWBRRUAiczyEN
/JIBWmVuatppwNcJRTZl06qmIElqmoBTWLZj0VmXj/+2SwVQUTNGNGsu0zzzTDS11rmeF1d1RcbKVvNcxtRH4ysh5JdIp1fn7G3l4CaTjJHBHo2Ufu0ebe3dfqqRc6QzmEr
/OfgpiDpcA7fxITX0pDokm+WaQMAZ7s6DEmgW+h7KL6ub0hVewzukba5dpvbyqycioDaomD4ywVa15csvmubwSRIALRH80uew0JcyeJSfEY8fS1Fud0BlG934DtI4HnT2CBM8C
/NKL76fLqPRGActEV4x0nvceNWm28oApI0hYpPUTv5YIP5Y=</wsse:BinarySecurityToken>
5    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-4bbe4224-d2df-4f57-814c-2b8a47ec328d">
6      <ds:SignedInfo>
7        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
8          <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap"/>
9        </ds:CanonicalizationMethod>
10       <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
11       <ds:Reference URI="#TS-91e2766f-c512-4440-bfa1-046bbdec9b7">
12         <ds:Transforms>
13           <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
14             <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap wsse"/>
15           </ds:Transform>
16         </ds:Transforms>

```

Fig. 3.32: Messaggio inviato dal fruttore

Informazioni Modl

Sicurezza Messaggio ID_AUTH_SOAP_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Bloccante

Sicurezza Messaggio

MessageID 428c7f0f-4fb2-11ed-a5ac-0242ac140002

WSA-From app1.enteesterno.govway.org

WSA-To luhnCheckerSoap.ente.govway.org

Expiration 2022-10-19_15:31:55.840

IssuedAt 2022-10-19_15:30:55.840

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Elementi SOAP Firmati

ReplyTo http://www.w3.org/2005/08/addressing

MessageID http://www.w3.org/2005/08/addressing

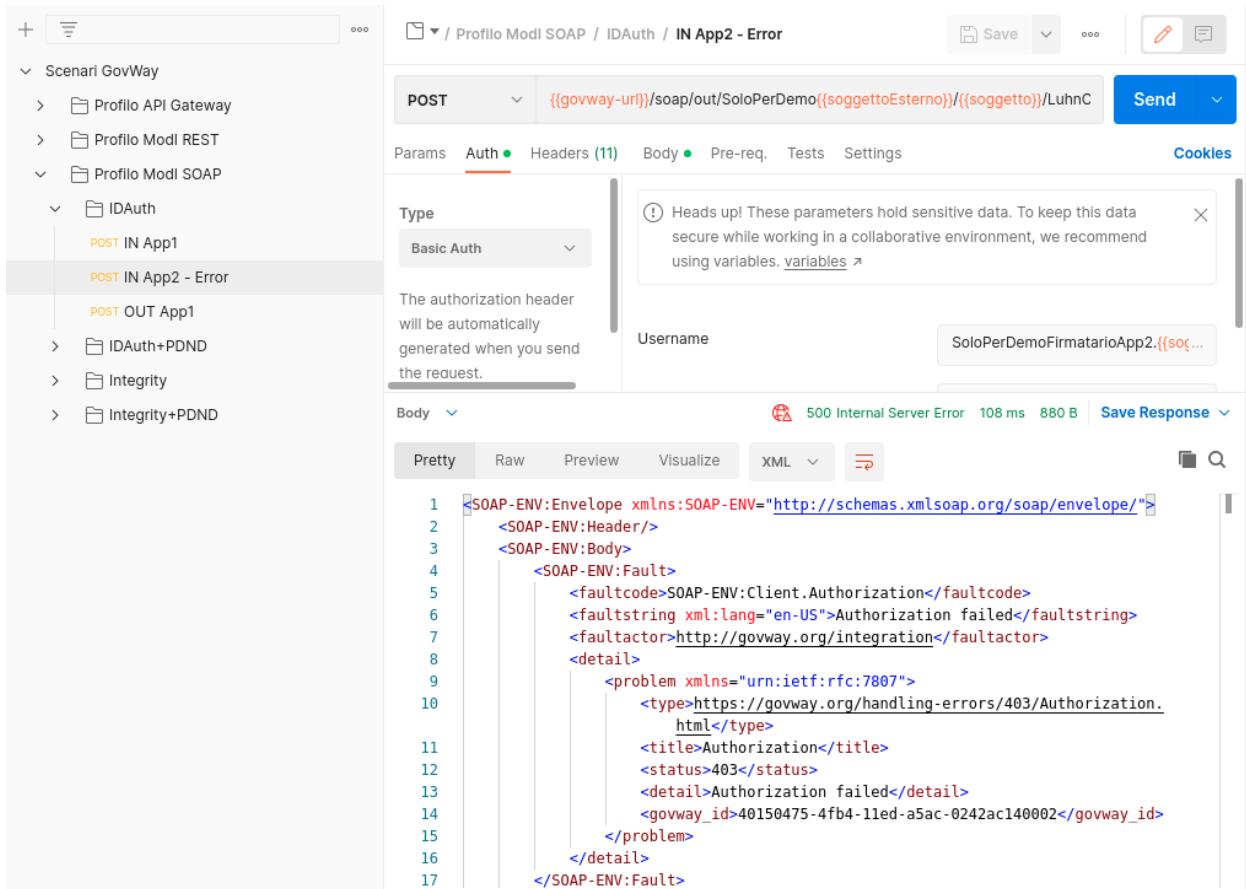
Action http://www.w3.org/2005/08/addressing

From http://www.w3.org/2005/08/addressing

To http://www.w3.org/2005/08/addressing

Timestamp http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd

Fig. 3.33: Traccia della richiesta elaborata dall'erogatore



The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Shows a tree structure of scenarios and profiles, including "Scenari GovWay", "Profilo API Gateway", "Profilo Modl REST", "Profilo Modl SOAP", "IDAuth", "IN App1", "IN App2 - Error", "OUT App1", "IDAuth+PDND", "Integrity", and "Integrity+PDND".
- Request Details:**
 - Method:** POST
 - URL:** {{govway-uri}}/soap/out/SoloPerDemo{{soggettoEsterno}}/{{soggetto}}/LuhnC
 - Auth:** Basic Auth (selected)
 - Headers:** (11) (shown in red)
 - Body:** (shown in red)
 - Tests:** (shown in red)
 - Settings:** (shown in red)
 - Cookies:** (shown in red)
- Right Panel:**
 - Type:** Basic Auth
 - Message:** Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [variables](#)
 - Username:** SoloPerDemoFirmatarioApp2.{{sog...}}
 - Body:** (shown in red)
 - Response:** 500 Internal Server Error 108 ms 880 B [Save Response](#)
 - Code View:**

```

1  <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
2    <SOAP-ENV:Header/>
3    <SOAP-ENV:Body>
4      <SOAP-ENV:Fault>
5        <faultcode>SOAP-ENV:Client.Authorization</faultcode>
6        <faultstring xml:lang="en-US">Authorization failed</faultstring>
7        <faultactor>http://govway.org/integration</faultactor>
8        <detail>
9          <problem xmlns="urn:ietf:rfc:7807">
10            <type>https://govway.org/handling-errors/403/Authorization.html</type>
11            <title>Authorization</title>
12            <status>403</status>
13            <detail>Authorization failed</detail>
14            <govway_id>40150475-4fb4-11ed-a5ac-0242ac140002</govway_id>
15          </problem>
16        </detail>
17      </SOAP-ENV:Fault>

```

Fig. 3.34: Pattern IDAuth - Erogazione API SOAP - Autorizzazione negata, esecuzione da Postman

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.35: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Configurazione*. Nel seguito sono evidenziate le sole differenze.

L’interfaccia wsdl del servizio soap è ottenibile all’indirizzo “<https://ws.cdyne.com/creditcardverify/luhnchecker.asmx?wsdl>”.

Registrazione API

Viene registrata l’API «CreditCardVerificationAuth» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «ID_AUTH_SOAP_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.36).

Erogazione

Si registra l’erogazione SOAP “LuhnCheckerSoapAuth”, relativa all’API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.37). In questo contesto vengono inseriti i dati necessari per validare le richieste in ingresso.

La sezione «ModI Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza da inserire nel messaggio di risposta (Fig. 3.38).

3.1.4 Fruizione API SOAP

Obiettivo

Fruire di un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza “ID_AUTH_SOAP_01” descritto nella sezione modipa_idar01.

Sintesi

Mostriamo in questa sezione come procedere per l’integrazione di un applicativo con un servizio SOAP erogato nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede il trust del certificato X.509 in modo da assicurare sia a livello di canale che a livello di messaggio l’autenticazione e autorizzazione del fruitore.

API > CreditCardVerificationAuth v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

Modi

Sicurezza Canale

Pattern Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern Direct Trust con certificato X.509

Applicabilità

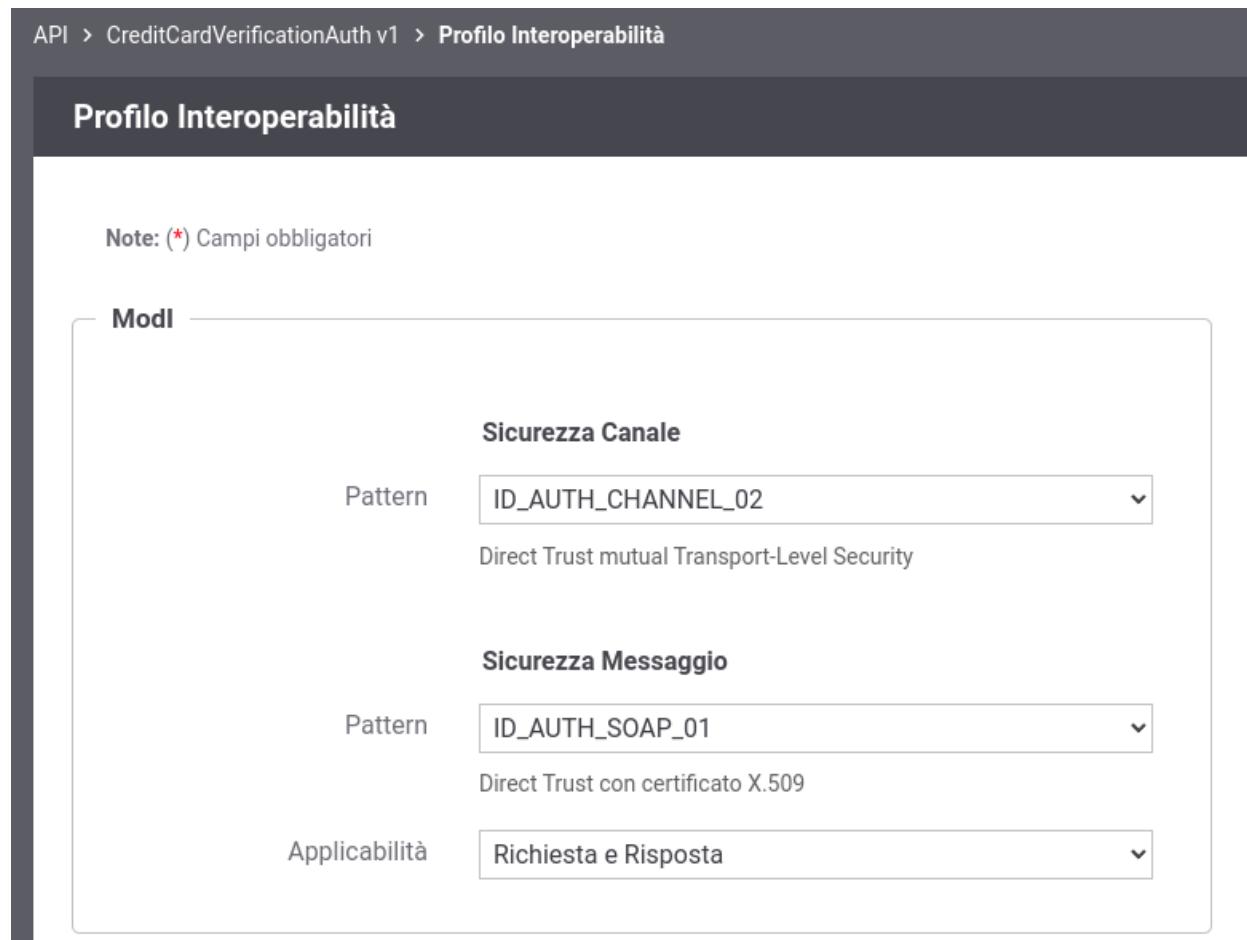


Fig. 3.36: Configurazione Pattern ModI «ID_AUTH_SOAP_01» sulla API SOAP

Modi - Richiesta

Sicurezza Messaggio

TrustStore Certificati

Time to Live

WSAddressing To

Se non viene fornito un valore, il valore atteso all'interno del security token corrisponderà all'url di invocazione

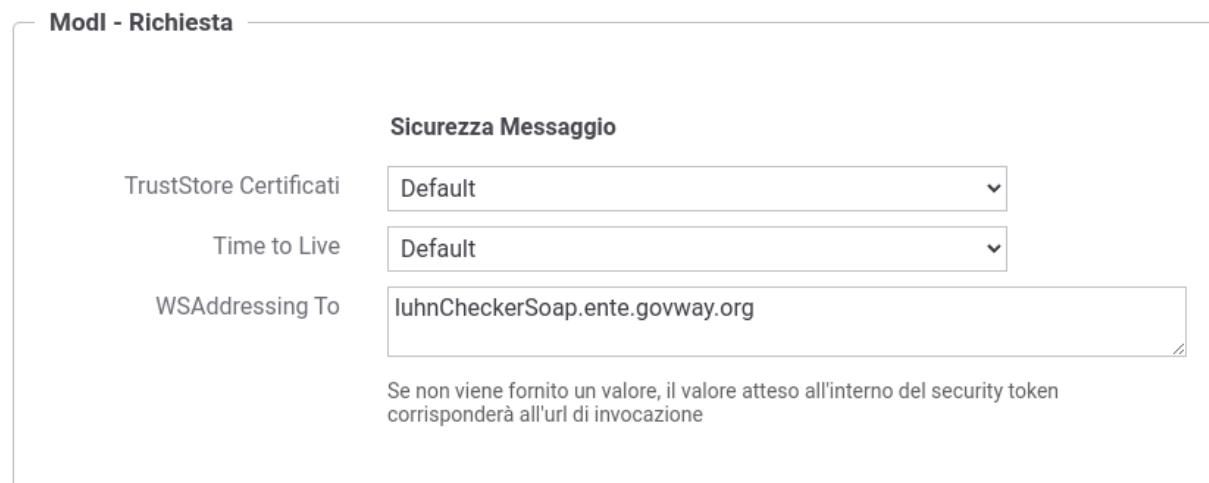


Fig. 3.37: Configurazione richiesta dell'erogazione

Modi - Risposta

Sicurezza Messaggio

Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
KeyStore	Default
Time to Live (secondi) *	60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Fig. 3.38: Configurazione risposta dell'erogazione

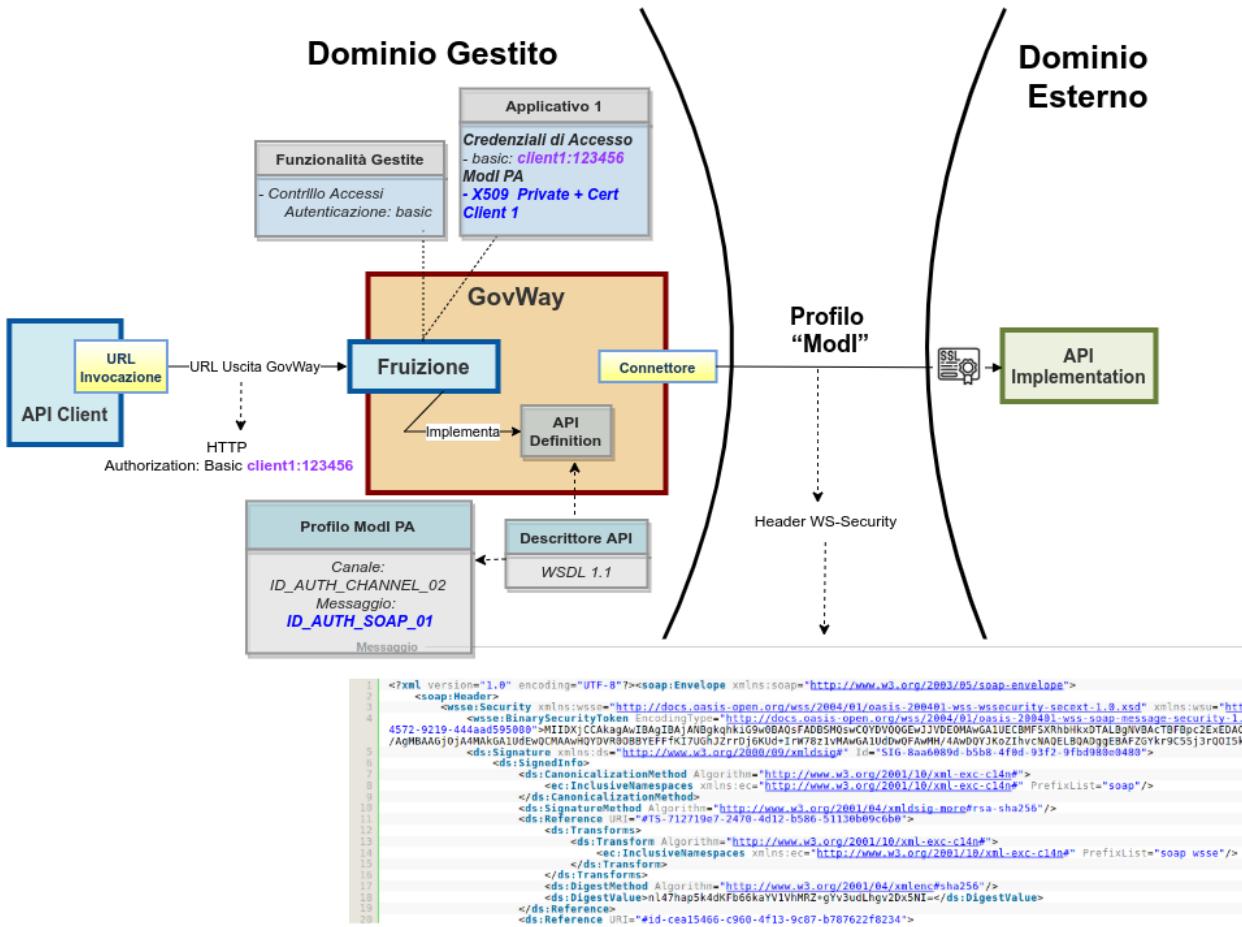


Fig. 3.39: Fruitione di una API SOAP con profilo "Modi", pattern ID_AUTH_SOAP_01

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_SOAP_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.40: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API SOAP di esempio (Credit Card Verification) definita con pattern di interazione Bloccante e pattern di sicurezza «ID_AUTH_CHANNEL_02» e «ID_AUTH_SOAP_01»;
- un'istanza Govway per la gestione del profilo ModI nel dominio del fruitore;
- un client del dominio gestito che invoca l'azione di esempio «CheckCC» sulla fruizione configurata su Govway.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

1. Il messaggio di richiesta inviato dal fruitore viene elaborato da Govway che, tramite la configurazione della firma digitale associata all'applicativo mittente, è in grado di produrre l'header WS-Security da inserire nella richiesta inviata all'erogatore. Da govwayMonitor si può visualizzare il messaggio di richiesta in uscita, analogo a quanto già visto in [Fig. 3.32](#).
2. Per verificare l'utilizzo del canale SSL, in accordo al pattern «ID_AUTH_CHANNEL_02», si procede come già illustrato per [Esecuzione](#).
3. Govway riceve la risposta dell'erogatore, dalla quale estrae il token di sicurezza al fine di effettuare i relativi controlli di validità e conservare la traccia. Consultando la traccia relativa alla trasmissione della risposta, sono visibili tra le altre informazioni i dati di autenticazione dell'erogatore e i riferimenti temporali.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in [Esecuzione](#).

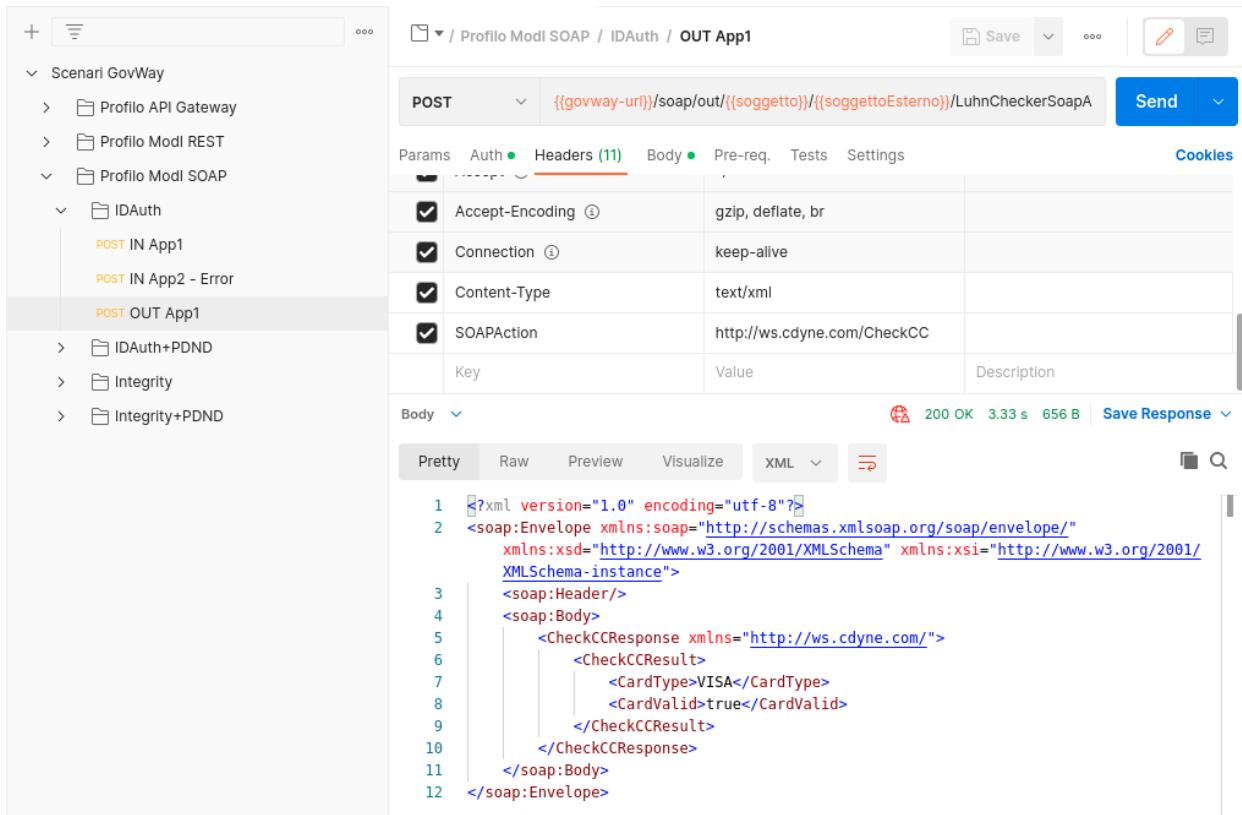


Fig. 3.41: Pattern IDAuth - Fruizione API SOAP, esecuzione da Postman

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.42: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Configurazione*. Nel seguito sono evidenziate le sole differenze.

Registrazione API

Viene registrata l’API «CreditCardVerificationAuth» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «ID_AUTH_SOAP_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.36).

Fruizione

Si registra la fruizione SOAP “LuhnCheckerSoapAuth”, relativa all’API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.44).

La sezione «ModI Risposta» definisce i criteri per la validazione dei messaggi di risposta (Fig. 3.45).

3.2 Pattern “INTEGRITY”

Gli scenari riportati in questa sezione riguardano API configurate con pattern modipa_idar03.

3.2.1 Erogazione API REST

Obiettivo

Esporre un servizio, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza “INTEGRITY_REST_01” descritto nella sezione modipa_idar03.

Sintesi

Mostriamo in questa sezione come procedere per l’esposizione di un servizio REST da erogare nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede oltre a garantire l’autenticazione dell’interlocutore un supporto a garanzia dell’integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

API > CreditCardVerificationAuth v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern ▼
Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern ▼
Direct Trust con certificato X.509

Applicabilità ▼

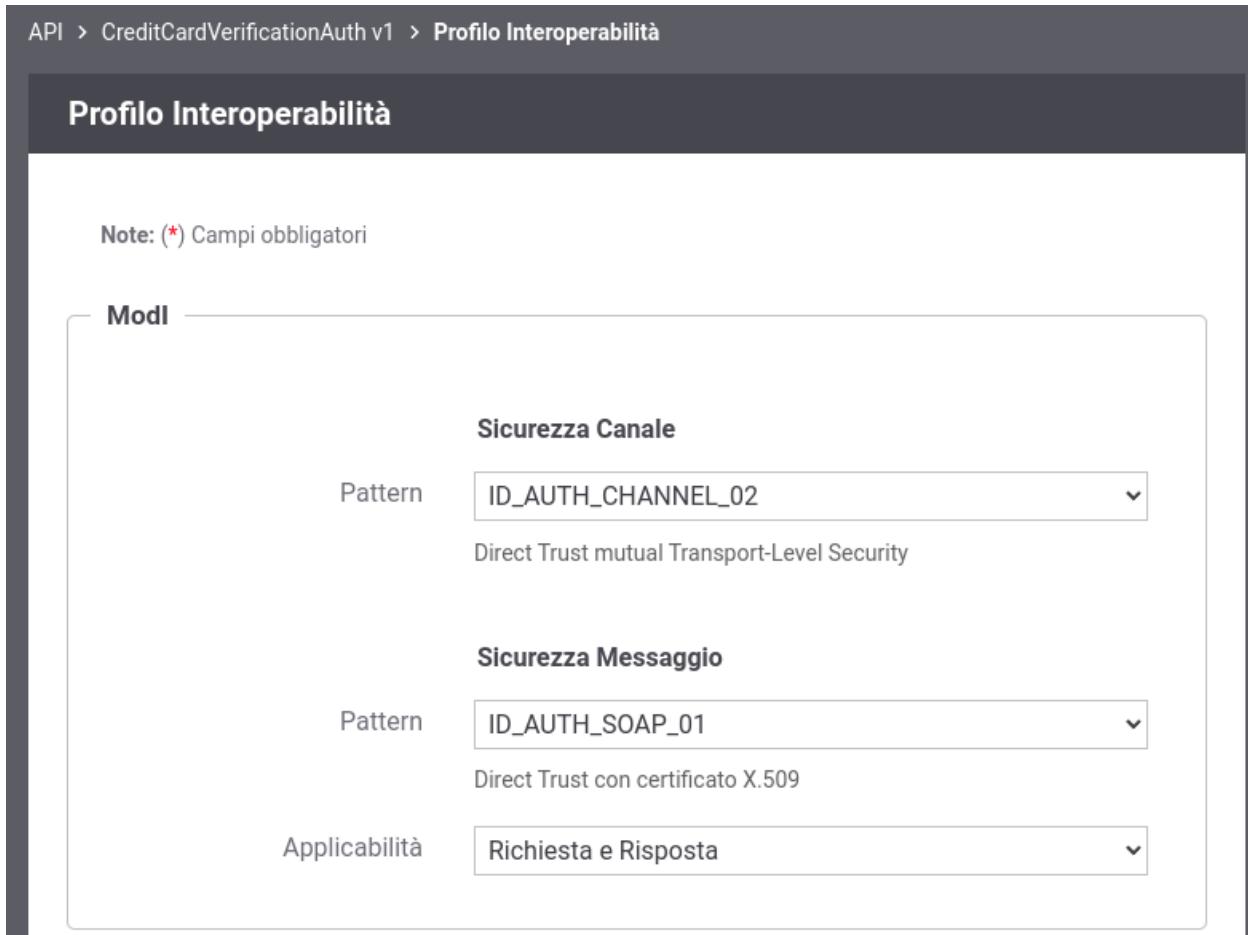


Fig. 3.43: Configurazione Pattern ModI «ID_AUTH_SOAP_01» sulla API SOAP

Modi - Richiesta

Sicurezza Messaggio

Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token

WSAddressing To ⓘ

Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore

Fig. 3.44: Configurazione richiesta della fruizione

Modi - Risposta

Sicurezza Messaggio

TrustStore Certificati	Default
Time to Live	Default

Verifica WSAddressing To La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente

ⓘ

Fig. 3.45: Configurazione risposta della fruizione

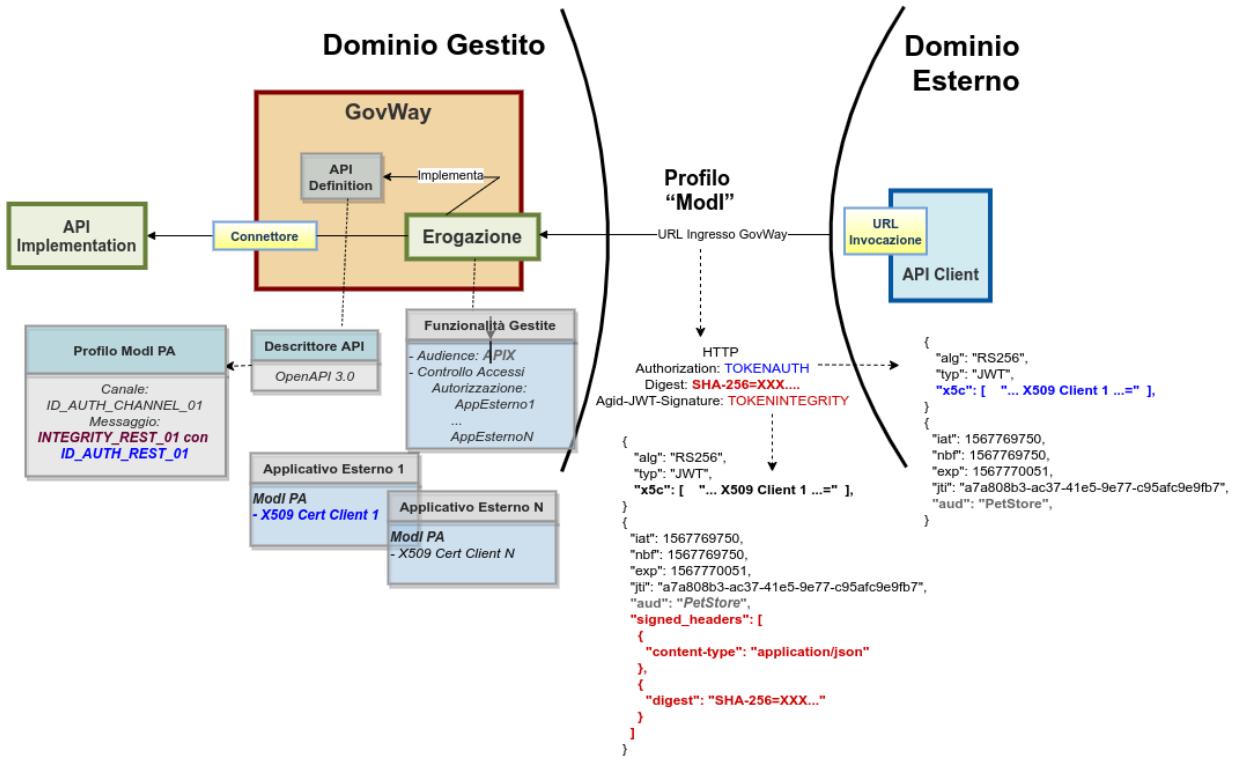


Fig. 3.46: Erogazione di una API REST con profilo "ModI", pattern INTEGRITY_REST_01 con ID_AUTH_REST_01

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01»
4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_REST_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.47: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza utilizzato che in questo scenario è «INTEGRITY_REST_01 con ID_AUTH_REST_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Fig. 3.48: Pattern Integrity - Erogazione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruitore, come in Fig. 3.49. Come si nota, al payload JSON è associato un insieme di header HTTP tra i quali «Authorization» e «Agid-Jwt-Signature» che contengono rispettivamente il token di sicurezza per l'autenticazione e per l'integrità. È inoltre presente l'header http «Digest» che contiene il valore per la verifica dell'integrità del payload.
- Grazie alle configurazioni presenti nell'erogazione, ed in particolare alla relazione di trust stabilita con il fruitore, Govway è in grado di validare i dati di sicurezza ricevuti andando a decodificare il token. Nella fase di validazione del token si può notare come la sezione header (Fig. 3.50) di entrambi i token «Authorization» e «Agid-Jwt-Signature» riportano l'identità del fruitore e il suo certificato X.509.
- Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruitore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso e la corrispondenza del digest con il payload e con il valore interno al token «Agid-Jwt-Signature». Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Le evidenze del processo di validazione sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.53). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dai token di sicurezza presenti, tra cui si può notare il digest e gli header firmati.

Headers	
Nome	
Content-Type	application/json
Govway-Message-Id	d1b37101-4fbb-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	0ab5fecb-2b64-497f-9a8e-ff0a6dbd24ab
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVClsImtpZC16ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6xWqdhfvHBaJT3on7jaCV6LVEXEaqAqfDWwI48L8SextE3UyuuGh-1s-g6320H8j6SIFr8tzsK4p-Fc94WcIxhMJxjXAer6Sh80
Agid-Jwt-Signature	eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVClsImtpZC16ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6WyJNSilVuNpGcBUWGoh1dKhKCv6nd6LFjWiFSdExxjto5i8lBtyjExSu06IHL0iaD2p1jkYrG37MgE6f-1xBYCqjElCchD6GQ8R4fEc5
Digest	SHA-256=OhjWochHmyIM/B4HeXiplNxygvqU7zKjERTUMDPVfhPY=
Accept	/*
Govway-Transaction-Id	d1a3b973-4fbb-11ed-a5ac-0242ac140002
Transfer-Encoding	chunked

Fig. 3.49: Messaggio inviato dal fruttore

```

HEADER: ALGORITHM & TOKEN TYPE

ID  {
  "alg": "RS256",
  "typ": "JWT",
  "kid": "app1.enteesterno.govway.org",
  "x5c": [
    "MIIE/jCCAuagAwIBAgICAPgwDQYJKoZIhvcNAQELBQAwNjELMAkGA1
    UEBhMCAxQxEzARBgNVBAoMCmdvdndheS5vcmcxEjAQBgNVBAMMCUdvd
    1dheSBDQTAeFw0yMjEwMTkwNzU1NThaFw0zNzEwMTUwNzU1NThaMEgx
    CzAJBgNVBAYTAm10MRMwEQYDVQQDApnb3Z3YXkub3JnMSQwIgYDVQQ
    DDBthcHAxLmVudGVFc3R1cm5vLmdvdndheS5vcmcwggEiMA0GCSqGSI
    b3DQEBAQUAA4IBDwAwggEKAoIBAQC1/cfENX06hdvEVxJiJAF00ePjn
    5Sh/HIJ2du8hRv0zA+KFFieaF4xh1mS0T1oq/vwdxFqvcd2k1bTJ37r
    jBo6DKuQZor83j/Do87x3sFJe/epGKx96Q3PRE9mA1qx3Y5FFShfGNZ
    zG2RFNA2jhVQ/bs8d9E051FC3XshF90CtJJs9LGvT2+0+uJK3siA6ht
    KcYQ58UcK1W1Y109MnXqaz82TiH93eTSkk33w0A9atzC0w3JAVmcRRk
    d0hFBjMeEvNR86cdNfy9Xit7ZDR11IB8tel0/f1/oAW0oK/3TbF1XOr
    VL1QhMc1JdqS3NwJLAyoqmZT/Xh5DqjDi7ldghwbAgMBAAGjggECMIH
    /MAkGA1UdEwQCMAAwEQYJYIZIAYb4QgEBBAQDAgeAMDMGCWCGSAGG+E
    IBDQQmFiRPcGVuU1NMIEd1bmVyYXR1ZCBDbG11bnQgQ2VydG1maWNhd
    GUwHQYDVR0OBYEFCBwk8Bs9JS+6c/vTU+JX0eqX81BMGYGA1UdIwRf
    MF2AFCqHFNpm2RdIA3igRXzNEeJ5ivegoTqk0DA2MQswCQYDVQQGEwJ
    pdDETMBEGA1UECgwKZ292d2F5Lm9yZzESMBAGA1UEAwJR292V2F5IE
    NBggkA4tGAdmeSJF4wDgYDVR0PAQH/BAQDAgXgMBMGA1UdJQQMMAoGC
    CsGAQUFBwMCMA0GCSqGSIb3DQEBCwUAA4ICAQDRj52cdYwcqFDNmC29
    CY0DR0N0TM/5RKq9sL6sgI7z4cUmkyIeGh/9YQDoRFhDBVGZ80rx0ka
    sZ/Po0Iuw+41f9IDTBe04Ym0CK3M1M9H2LiEKe9hngRtjzGw5tFRQnq
    KbLLX61otJAXuE488SrSAMbEdez1bZt+V1Sgc48f0KsjShUs8CwSW0G
    6RE5w4Q4oa0dX971PTziWDoFnxBfN17/HAYA0625/vcp8PrZLqhTIGH
    7dt+1T4Hb+i10wKBS7B8Cab0Gh0spiHDDgNEYX50d1ZYmWJQ10ysK6
    1Yx1WtCrKPfmsvSeqiVxJPHUgwTsFPrgoVRt+dT1NnAdXYxFk0Yxz7z
    n7qeKDi6cXHLTsYet1cQfedyDPE0rli4GFL1KY37NFqRtJx5NadkJk6
    GXk43zIFQo119PGJ8nVHupB6IBJ1h/6xem1TTMSt52zcjV4b5zRHL8Z
    qNF+S0QnJKcH2FcyAYuGjuVj0qa5rhi5wNcy7ilcDShM8tsPJ5qpW1M
    E0mmhWWY+w5KBCpMoLBn9cvqAn/N19L3e3SqH1Klp8Hw05CtH4/tLEe
    3N+0z+8xzcmLdqbaZ9nD7YVLVoyt5Y+Ixuj17F18dzEh9dzLhJojsB
    mPjoFMMyulbpjZG0A1TjKVpkxyXgaqsd9Hjs4ATg79Vk8U/GnEXJhXQ
    xU2TYw=="
  ],
  "x5t#S256": "agRQxqs-
  VYDP2NIzbR7XH2GiInWH2bcL1xMPhimfMKK"
}

```

Fig. 3.50: Sezione «Header» del Token di sicurezza «Authorization» e «Agid-Jwt-Signature»
 I payload dei due token invece differiscono (Fig. 3.51 e Fig. 3.52). In entrambi sono presenti i riferimenti temporali (iat, nbf, exp)
 e l'audience (aud), mentre solamente nel payload del token «Agid-Jwt-Signature» è presente il claim «signed_headers» utilizzato
 per la verifica dell'integrità.

```

PAYLOAD: DATA

{
  "iat": 1666176318,
  "nbf": 1666176318,
  "exp": 1666176378,
  "jti": "1f46c4b4-4f9b-11ed-a5ac-0242ac140002",
  "aud": "petstore.ente.govway.org",
  "client_id": "app1.enteesterno.govway.org",
  "iss": "SoloPerDemoEnteEsterno",
  "sub": "SoloPerDemoFirmatarioApp1"
}

```

Fig. 3.51: Sezione «Payload» del Token di sicurezza «Authorization»

```

PAYLOAD: DATA

{
  "iat": 1666190361,
  "nbf": 1666190361,
  "exp": 1666190421,
  "jti": "d1b37101-4fbb-11ed-a5ac-0242ac140002",
  "aud": "petstore.ente.govway.org",
  "client_id": "app1.enteesterno.govway.org",
  "iss": "SoloPerDemoEnteEsterno",
  "sub": "SoloPerDemoFirmatarioApp1",
  "signed_headers": [
    {
      "digest": "SHA-256=0hjWocHmy1M/B4HeXlplNxygvqU7zKjERTUMDPVfhPY="
    },
    {
      "content-type": "application/json"
    }
  ]
}

```

Fig. 3.52: Sezione «Payload» del Token di sicurezza «Agid-Jwt-Signature»

Informazioni Mod

Sicurezza Messaggio INTEGRITY_REST_01 con ID_AUTH_REST_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Accesso CRUD

Sicurezza Messaggio

Digest SHA-256=OhjWocHmyIM/B4HeXlplNxygvqU7zKjERTUMDPVfhPY=

ClientId app1.enteesterno.govway.org

Subject SoloPerDemoFirmatarioApp1

Issuer SoloPerDemoEnteEsterno

MessageId d1b37101-4fb-11ed-a5ac-0242ac140002

Audience petstore.ente.govway.org

NotBefore 2022-10-19_16:39:21.000

Expiration 2022-10-19_16:40:21.000

IssuedAt 2022-10-19_16:39:21.000

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Headers HTTP Firmati

content-type application/json

digest SHA-256=OhjWocHmyIM/B4HeXlplNxygvqU7zKjERTUMDPVfhPY=

Fig. 3.53: Traccia della richiesta elaborata dall'erogatore

- Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo "App1-ModI" identificato grazie al certificato X.509 presente all'interno del token. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

The screenshot shows the Postman interface with the following details:

- Request URL:** POST /IN App2 - Error
- Request Headers:** {{govway-url}}/rest/out/SoloPerDemo{{soggettoEsterno}}/{{soggetto}}/PetStore
- Query Params:** Key: Value
- Body:** JSON response (Pretty):


```

1  {
2    "type": "https://govway.org/handling-errors/403/Authorization.html",
3    "title": "Authorization",
4    "status": 403,
5    "detail": "Authorization failed",
6    "govway_id": "6072f3df-4fbe-11ed-a5ac-0242ac140002"
7  }
      
```
- Response Headers:** 403 Forbidden, 46 ms, 446 B

Fig. 3.54: Pattern Integrity - Erogazione API REST - Autorizzazione negata, esecuzione da Postman

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. La trasmissione è basata sul pattern «ID_AUTH_CHANNEL_02», riguardo la sicurezza canale, come evidenziato nei messaggi diagnostici dalla presenza degli elementi dell'handshake SSL e relativi dati dei certificati scambiati (Fig. 3.5).
2. La sicurezza messaggio applicata è quella dei pattern «ID_AUTH_REST_02» e «INTEGRITY_REST_01» come ampiamente mostrato nelle tracce dei messaggi di richiesta e risposta, dove sono presenti i certificati degli applicativi e le firme dei payload (e le relative validazioni).
3. L'identificazione del fruitore avviene rispetto al certificato X.509 presente all'interno del token. È stato anche mostrato come sia possibile configurare criteri autorizzativi puntuali.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità "ModI". Si suggerisce inoltre di selezionare il soggetto "Ente" per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.55: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario *Configurazione* con le sole differenze dovute al differente pattern di sicurezza utilizzato «INTEGRITY_REST_01 con ID_AUTH_REST_01».

Registrazione API

Viene registrata l'API «PetStoreIntegrity» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «INTEGRITY_REST_01 con ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.56).

API > PetStoreIntegrity v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_02

Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern: INTEGRITY_REST_01 con ID_AUTH_REST_01

Integrità payload del messaggio

Header HTTP del Token: Agid-JWT-Signature + Authorization Bearer

Applicabilità: Richiesta e Risposta

Digest Richiesta: Non ripudiabilità della trasmissione (i)

Informazioni Utente: Dati dell'utente che effettua la richiesta (i)

Fig. 3.56: Configurazione Pattern ModI «INTEGRITY_REST_01 con ID_AUTH_REST_01» sulla API REST

Erogazione

Si registra l'erogazione «PetStoreIntegrity», relativa all'API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.57). In questo contesto vengono inseriti i dati necessari per validare le richieste in ingresso.

Fig. 3.57: Configurazione richiesta dell'erogazione

La sezione «ModI Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza da inserire nel messaggio di risposta (Fig. 3.58).

3.2.2 Fruizione API REST

Obiettivo

Fruire di un servizio REST, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza “INTEGRITY_REST_01” descritto nella sezione modipa_idar03.

Sintesi

Mostriamo in questa sezione come procedere per l'integrazione di un applicativo con un servizio REST erogato nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede oltre a garantire l'autenticazione dell'interlocutore un supporto a garanzia dell'integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»

Modi - Risposta

Sicurezza Messaggio

Algoritmo: RS256

HTTP Headers da firmare *: Digest x, Content-Type x, Content-Encoding x

Riferimento X.509: Utilizza impostazioni della Richiesta

Certificate Chain:

KeyStore: Default

Time to Live (secondi) *: 60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Claims: ⓘ

Indicare per riga i claims (nome=valore); visualizzare 'info' per maggiori dettagli

▼ Contemporaneità Token Authorization e Agid-JWT-Signature

Fig. 3.58: Configurazione risposta dell'erogazione

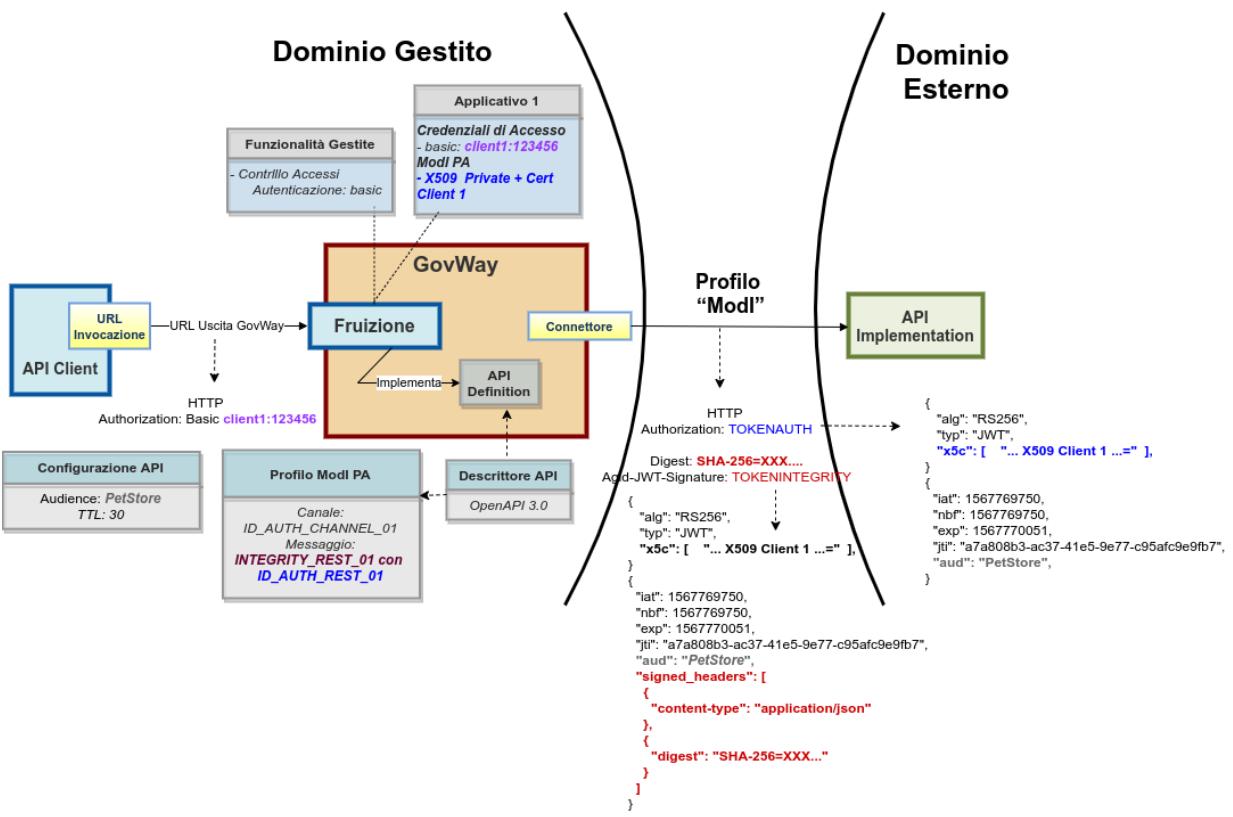


Fig. 3.59: Fruizione di una API REST con profilo “ModI”, pattern INTEGRITY_REST_01 con ID_AUTH_REST_01

3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01»
4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_REST_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.60: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza utilizzato che in questo scenario è «INTEGRITY_REST_01 con ID_AUTH_REST_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

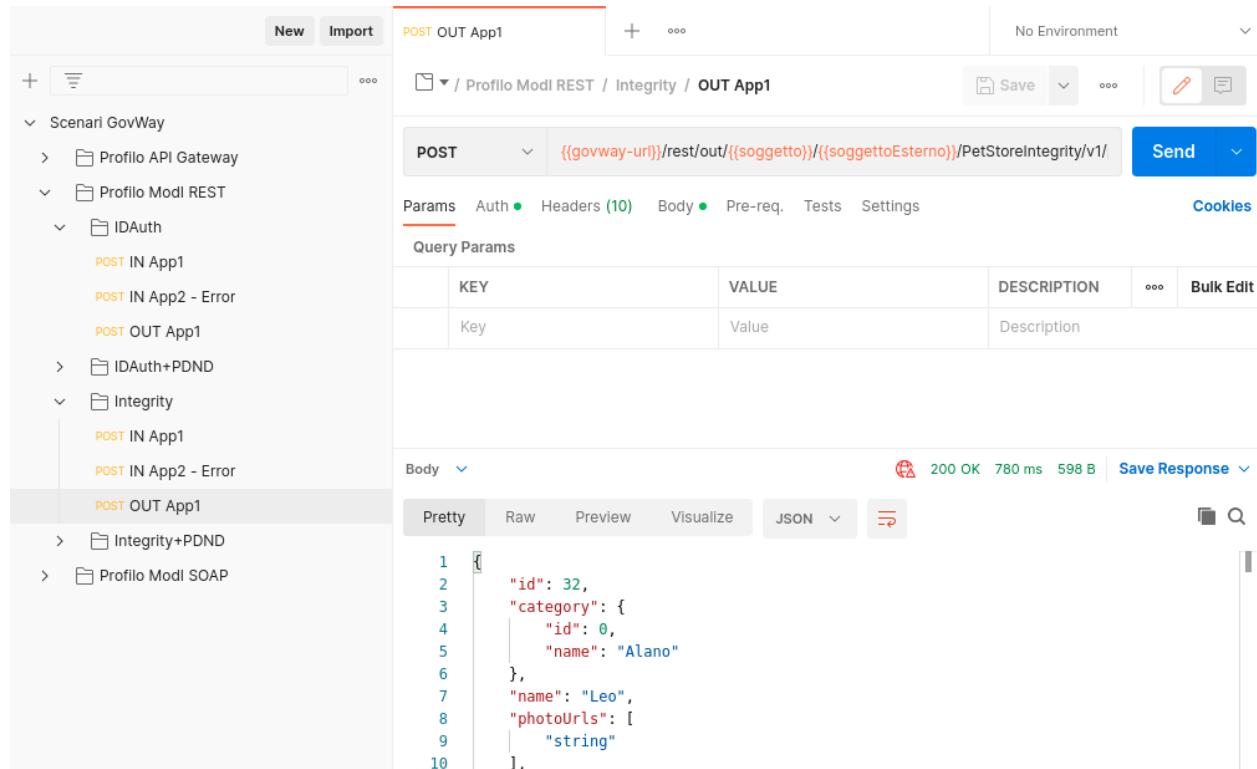


Fig. 3.61: Pattern Integrity - Fruizione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto nelle diverse fasi dell'esecuzione andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Il messaggio di richiesta inviato dal fruitore viene elaborato da Govway che, tramite la configurazione della firma digitale associata all'applicativo mittente, è in grado di produrre il token di sicurezza da inviare con la richiesta all'erogatore. Da govwayMonitor si può visualizzare il messaggio di richiesta in uscita che è il medesimo di quello in entrata con la differenza che sono stati aggiunti gli header HTTP «Authorization» e «Agid-Jwt-Signature» che contengono rispettivamente il token di sicurezza per l'autenticazione e per l'integrità. È inoltre presente l'header http «Digest» che contiene il valore utilizzabile dall'erogatore per la verifica dell'integrità del payload. (Fig. 3.62).

Headers	
Nome	
Content-Type	application/json
Govway-Message-Id	d1b37101-4fbb-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	0ab5fecb-2b64-497f-9a8e-ff0a6dbd24ab
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6xWqdhfvHBaJT3on7jaCV6LVEXEaqAqfDWwI48L8SextE3UyuuGh-1s-g632OH8j6SIf8tzsK4p-Fc94WclxhMJxjXAer6Sh80
Agid-Jwt-Signature	eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6WyJNSjliVuNpGcBUWGoh1dKhKCv6nd6LFjWIFsdExxjto5i8iBtyjExSu06IHL0iaD2pI1jkYrG37MgE6f-1xBYCqlEIcchD6GQ8R4fEc5
Digest	SHA-256=OhJwochmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=
Accept	*/*
Govway-Transaction-Id	d1a3b973-4fbb-11ed-a5ac-0242ac140002
Transfer-Encoding	chunked

Fig. 3.62: Messaggio di richiesta in uscita (con token di sicurezza inseriti nell'header HTTP)

- L'header e i payload dei token sono identici a quelli visualizzati nello scenario di erogazione REST, relativamente al messaggio in ingresso (Fig. 3.50, Fig. 3.51 e Fig. 3.52). Le informazioni inserite nel token vengono anche tracciate e sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta

(Fig. 3.63). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dai token di sicurezza, tra cui si può notare il digest e gli header http firmati.

Informazioni ModI

Sicurezza Messaggio INTEGRITY_REST_01 con ID_AUTH_REST_01

Sicurezza Canale ID_AUTH_CHANNEL_01

Interazione Accesso CRUD

Sicurezza Messaggio

X509-Issuer	CN=GovWay CA, O=govway.org, C=it
X509-Subject	CN=app1.ente.govway.org, O=govway.org, C=it
Digest	SHA-256=0hjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=
Subject	App1-PDND
Issuer	Ente
Clientid	Ente/App1-PDND
Audience	petstore.enteEsterno.govway.org
MessageId	d59e4915-508b-11ed-a5ac-0242ac140002
Expiration	2022-10-20_17:29:23.000
NotBefore	2022-10-20_17:28:23.000
IssuedAt	2022-10-20_17:28:23.000

Headers HTTP Firmati

content-type	application/json
digest	SHA-256=0hjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=

Fig. 3.63: Traccia della richiesta generata dal fruitore

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. La trasmissione è basata sul pattern «ID_AUTH_CHANNEL_02», riguardo la sicurezza canale, come evidenziato nei messaggi diagnostici dalla presenza degli elementi dell'handshake SSL e relativi dati dei certificati scambiati (Fig. 3.23).
2. La sicurezza messaggio applicata è quella dei pattern «ID_AUTH_REST_02» e «INTEGRITY_REST_01», come ampiamente mostrato nelle tracce dei messaggi di richiesta e risposta, dove sono presenti i certificati degli applicativi e le firme dei payload (e le relative validazioni).

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.64: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario *Configurazione* con le sole differenze dovute al differente pattern di sicurezza utilizzato «INTEGRITY_REST_01 con ID_AUTH_REST_01».

Registrazione API

Viene registrata l’API «PetStoreIntegrity» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «INTEGRITY_REST_01 con ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.65).

Fruizione

Si registra la fruizione «PetStoreIntegrity», relativa all’API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.66). In particolare è possibile specificare l’audience atteso dall’erogatore e il tempo di validità del token.

La sezione «ModI Risposta» definisce i criteri per la validazione dei messaggi di risposta, come la posizione del token di sicurezza e il truststore per l’autenticazione dell’erogatore (Fig. 3.67).

3.2.3 Erogazione API SOAP

Obiettivo

Esportare un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza «INTEGRITY_SOAP_01» descritto nella sezione modipa_idar03.

Sintesi

Mostriamo in questa sezione come procedere per l’esposizione di un servizio SOAP da erogare nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede oltre a garantire l’autenticazione dell’interlocutore un supporto a garanzia dell’integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
3. L’autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_SOAP_01»

API > PetStoreIntegrity v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

Modi

Sicurezza Canale

Pattern ▼
Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern ▼
Integrità payload del messaggio

Header HTTP del Token ▼

Applicabilità ▼

Digest Richiesta Non ripudiabilità della trasmissione (i)

Informazioni Utente Dati dell'utente che effettua la richiesta (i)

Fig. 3.65: Configurazione Pattern ModI «INTEGRITY_REST_01 con ID_AUTH_REST_01» sulla API

Modi - Richiesta

Sicurezza Messaggio

Algoritmo	<input type="text" value="RS256"/>
HTTP Headers da firmare *	<input type="checkbox"/> Digest <input checked="" type="checkbox"/> Content-Type <input checked="" type="checkbox"/> Content-Encoding
Riferimento X.509	<input type="checkbox"/> x5c (Certificate) <input type="checkbox"/> x5t#256 (Certificate SHA-256 Thumbprint) <input type="checkbox"/> x5u (URL)
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	<input type="text" value="60"/>
Indica la validità temporale, in secondi, a partire dalla data di creazione del security token	
Audience	<input type="text" value="petstore.enteEsterno.govway.org"/> i
Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore	
Claims	<input type="text"/> i
Indicare per riga i claims (nome=valore); visualizzare 'info' per maggiori dettagli	
▼ Contemporaneità Token Authorization e Agid-JWT-Signature	

Fig. 3.66: Configurazione richiesta della fruizione

Modi - Risposta

Sicurezza Messaggio

Riferimento X.509	Utilizza impostazioni della Richiesta
TrustStore Certificati	Default
Time to Live	Default
Verifica Audience	<input checked="" type="checkbox"/> La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente

▼ Contemporaneità Token Authorization e Agid-JWT-Signature

Fig. 3.67: Configurazione risposta della fruizione

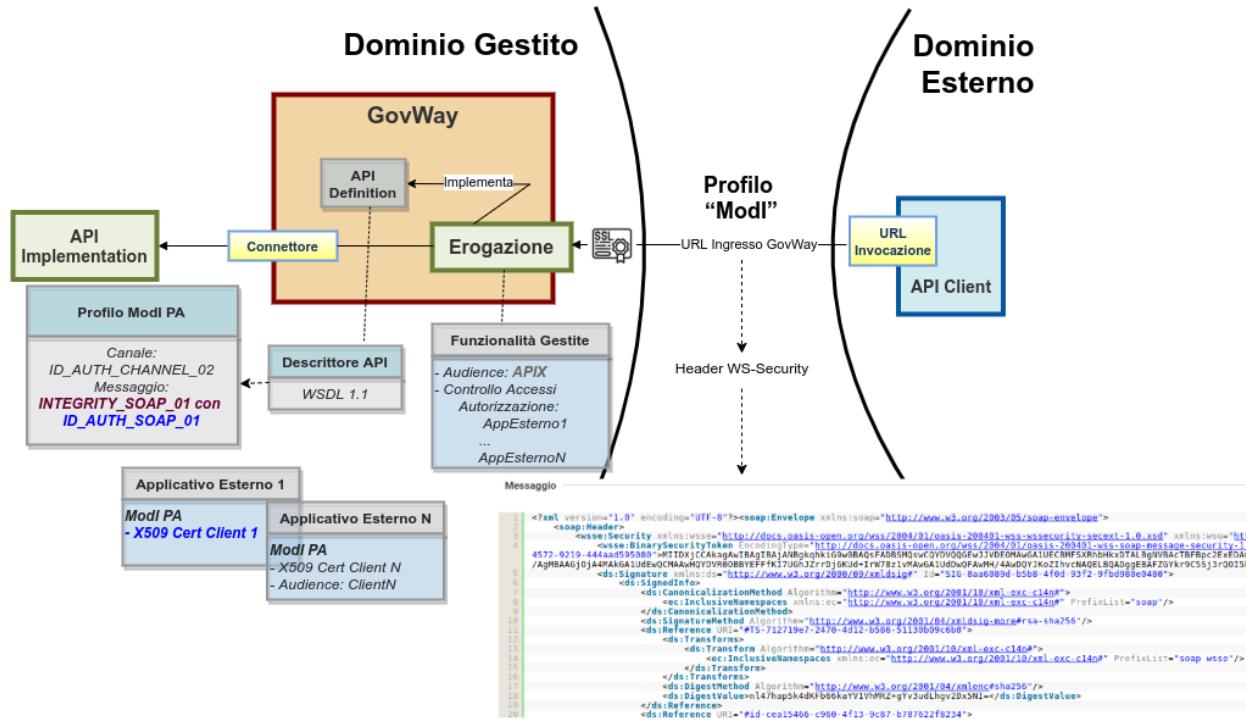


Fig. 3.68: Erogazione di una API SOAP con profilo "Modi", pattern INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01

4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_SOAP_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.69: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza utilizzato che in questo scenario è «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

 A screenshot of the Postman interface. The left sidebar shows a tree structure with categories like "Scenari GovWay", "Profilo API Gateway", "Profilo ModI REST", "Profilo ModI SOAP", "IDAuth", "Integrity", and specific requests like "POST IN App1", "POST IN App2 - Error", "POST OUT App1", and "IDAuth+PDND". The main panel shows a "POST IN App1" request. The "Headers" tab is selected, showing "Content-Type: application/json" and "Accept: application/json". The "Body" tab shows an XML payload for a Luhn check. The "Tests" tab contains a script to verify the response. The "Body" tab also shows a "Pretty" view of the XML code.


```

1  <?xml version="1.0" encoding="utf-8"?>
2  <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3      <soap:Header>
4          <CheckCCResponse xmlns="http://ws.cdyne.com/">
5              <CheckCCResult>
6                  <CardType>VISA</CardType>
7                  <CardValid>true</CardValid>
8              </CheckCCResult>
9          </CheckCCResponse>
10     </soap:Header>
11 </soap:Envelope>
  
```

Fig. 3.70: Pattern Integrity - Erogazione API SOAP, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console govwayMonitor.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruitore, come in Fig. 3.32. Come si nota, il messaggio SOAP contiene nell'header WS-Security sia il token di sicurezza (elemento «Binary-SecurityToken») sia l'elemento «WSAddressing - To» e il digest del payload (elemento «DigestValue») prodotti dal fruitore con la relativa firma digitale (elemento «SignatureValue»).

Messaggio

```

1  <?xml version="1.0" encoding="UTF-8"?><soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
2    <soap:Header>
3      <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#"
4        <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#"
c7761d94d64f">MIE/zCAuegAwIBAgICAN4wDQYJKoZIhvNAQELBQAwhjELMAkGA1UEBhMCaX0xEzARBgNVBaoCmndvdnhes55vcmcxEjAQBgNVBAMMCUdvlldheSB0QTAf
/Mu06/YXIV1DHLYMjypb/fL0SL8SKA6uW9swPXC0gJPK9aqgwiv0/Bw2Lpv1657H+BtN1e8fhsMnUnL7C25Hba/WivKh78213F5LYc4sY8H9nfC/fa6QUouidTxWhkKwzNl
/ZAJBgNVHRMEdjAAMBEggCwCGSAGG+EIBAQEAWiHgDA2BglghkgBvhvCAQ0EjYKt3B1bNTTCBHZw5LcmF0ZwQ0zpxpZw50IENLcnRpZmljYXRLMB0GA1udg0WBBrUAcYENj
/JIBWmVuatppwNcJRTZ106qmIElqmoBTWLZj0Mx1/+zSwvQUTMNGNsUzzzTDS11rmel1diRcbKvvNcxtrPHH4sh5jdIp1fN7G3l4CatjJHBH02Ufu0eb63dfqqRc6QzMr
/OFppiDpcA7fXITX0gD0km+WaqMAZ7s6DEmgW=h7KLk6ub0hVewzukbaSdpYbgycioDaom04ywva15csmubwSRIAlRH80uew03cyeJSfEY8fSlFudoBLG934DtI4HnT2CBM8
/NKL76fLQPRGAcHtEV4x0nvCe8NwM28oAp0hYpPUTv5YIP5Y=</wsse:BinarySecurityToken>
5    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-4bbe4224-d2df-4f57-814c-2b8a47ec328d">
6      <ds:SignedInfo>
7        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
8          <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap"/>
9        </ds:CanonicalizationMethod>
10       <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
11       <ds:Reference URI="#TS-91e2766f-c512-4440-bfa1-046bbdec9b7">
12         <ds:Transforms>
13           <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
14             <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap wsse"/>
15           </ds:Transform>
16         </ds:Transforms>

```

Fig. 3.71: Messaggio inviato dal fruitore

- Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruitore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso e la corrispondenza del digest rispetto al payload. Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Le evidenze del processo di validazione sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.72). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dal token di sicurezza presente nell'header soap.
- Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo “App1-ModI” identificato grazie al certificato X.509 presente all'interno dell'header WS-Security. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.

Informazioni Modelli

Sicurezza Messaggio INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Bloccante

Sicurezza Messaggio

MessageId 13526172-4fc9-11ed-a5ac-0242ac140002

WSA-From app1.enteesterno.govway.org

WSA-To luhnCheckerSoap.ente.govway.org

Digest SHA256=sRq5LjK63zpG/FhfMWb/IE1HtNE2w1XYhHdLIWgxuX0=

Expiration 2022-10-19_18:15:14.957

IssuedAt 2022-10-19_18:14:14.957

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Elementi SOAP Firmati

Body http://schemas.xmlsoap.org/soap/envelope/

ReplyTo http://www.w3.org/2005/08/addressing

MessageID http://www.w3.org/2005/08/addressing

Action http://www.w3.org/2005/08/addressing

From http://www.w3.org/2005/08/addressing

To http://www.w3.org/2005/08/addressing

Timestamp http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd

Fig. 3.72: Traccia della richiesta elaborata dall'erogatore

The screenshot shows the Postman application interface. On the left, a sidebar lists various scenarios and profiles, including 'Scenari GovWay', 'Profilo API Gateway', 'Profilo Modi REST', 'Profilo Modi SOAP', 'IDAuth', 'IN App1', 'IN App2 - Error', 'OUT App1', 'OUT App1', 'Integrity', 'IN App1', 'IN App2 - Error', 'OUT App1', and 'Integrity+PDND'. The main workspace is titled 'POST IN App2 - Error' and shows a request URL: `http://{{govway-uri}}/soap/out/SoloPerDemo{{soggettoEsterno}}/{{soggetto}}/LuhnC`. The 'Params' tab is selected, showing a single parameter 'Key' with 'Value' 'Value'. The 'Body' tab shows the XML response received from the server, which is a SOAP fault message. The XML content is as follows:

```

1  <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
2    <SOAP-ENV:Header/>
3    <SOAP-ENV:Body>
4      <SOAP-ENV:Fault>
5        <faultcode>SOAP-ENV:Client.Authorization</faultcode>
6        <faultstring xml:lang="en-US">Authorization failed</faultstring>
7        <faultactor>http://govway.org/integration</faultactor>
8        <detail>
9          <problem xmlns="urn:ietf:rfc:7807">
10            <type>https://govway.org/handling-errors/403/Authorization_
11              html</type>
12            <title>Authorization</title>
13            <status>403</status>
14            <detail>Authorization failed</detail>
15            <govway_id>47814c63-4fc9-11ed-a5ac-0242ac140002</govway_id>
16          </problem>
17        </detail>
18      </SOAP-ENV:Fault>

```

Fig. 3.73: Pattern Integrity - Erogazione API SOAP - Autorizzazione negata, esecuzione da Postman



Fig. 3.74: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Configurazione*. Nel seguito sono evidenziate le sole differenze.

L’interfaccia wsdl del servizio soap è ottenibile all’indirizzo “<https://ws.cdyne.com/creditcardverify/luhnchecker.asmx?wsdl>”.

Registrazione API

Viene registrata l’API «CreditCardVerificationIntegrity» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.75).

API > CreditCardVerificationIntegrity v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_02

Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern: INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01

Integrità payload del messaggio

Applicabilità: Richiesta e Risposta

Digest Richiesta: Non ripudiabilità della trasmissione i

Informazioni Utente: Dati dell’utente che effettua la richiesta i

Fig. 3.75: Configurazione Pattern ModI «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» sulla API SOAP

Erogazione

Si registra l’erogazione SOAP “LuhnCheckerSoapIntegrity”, relativa all’API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.76). In questo contesto vengono inseriti i dati necessari per validare le richieste in ingresso.

The screenshot shows a configuration interface for a SOAP request. At the top, a header reads 'ModI - Richiesta'. Below it, a section titled 'Sicurezza Messaggio' contains three dropdown menus: 'TrustStore Certificati' set to 'Default', 'Time to Live' set to 'Default', and 'WSAddressing To' set to 'luhnCheckerSoap.ente.govway.org'. A note below the dropdowns states: 'Se non viene fornito un valore, il valore atteso all'interno del security token corrisponderà all'url di invocazione'.

Fig. 3.76: Configurazione richiesta dell’erogazione

La sezione «ModI Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza da inserire nel messaggio di risposta (Fig. 3.77).

3.2.4 Fruizione API SOAP

Obiettivo

Fruire di un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza “INTEGRITY_SOAP_01” descritto nella sezione modipa_idar03.

Sintesi

Mostriamo in questa sezione come procedere per l’integrazione di un applicativo con un servizio SOAP erogato nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario, tra quelli prospettati nel Modello di Interoperabilità di AGID, che prevede oltre a garantire l’autenticazione dell’interlocutore un supporto a garanzia dell’integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_02»
3. L’autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_SOAP_01»
4. L’integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_SOAP_01»

Modi - Risposta

Sicurezza Messaggio

Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
KeyStore	Default
Time to Live (secondi) *	60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Fig. 3.77: Configurazione risposta dell'erogazione

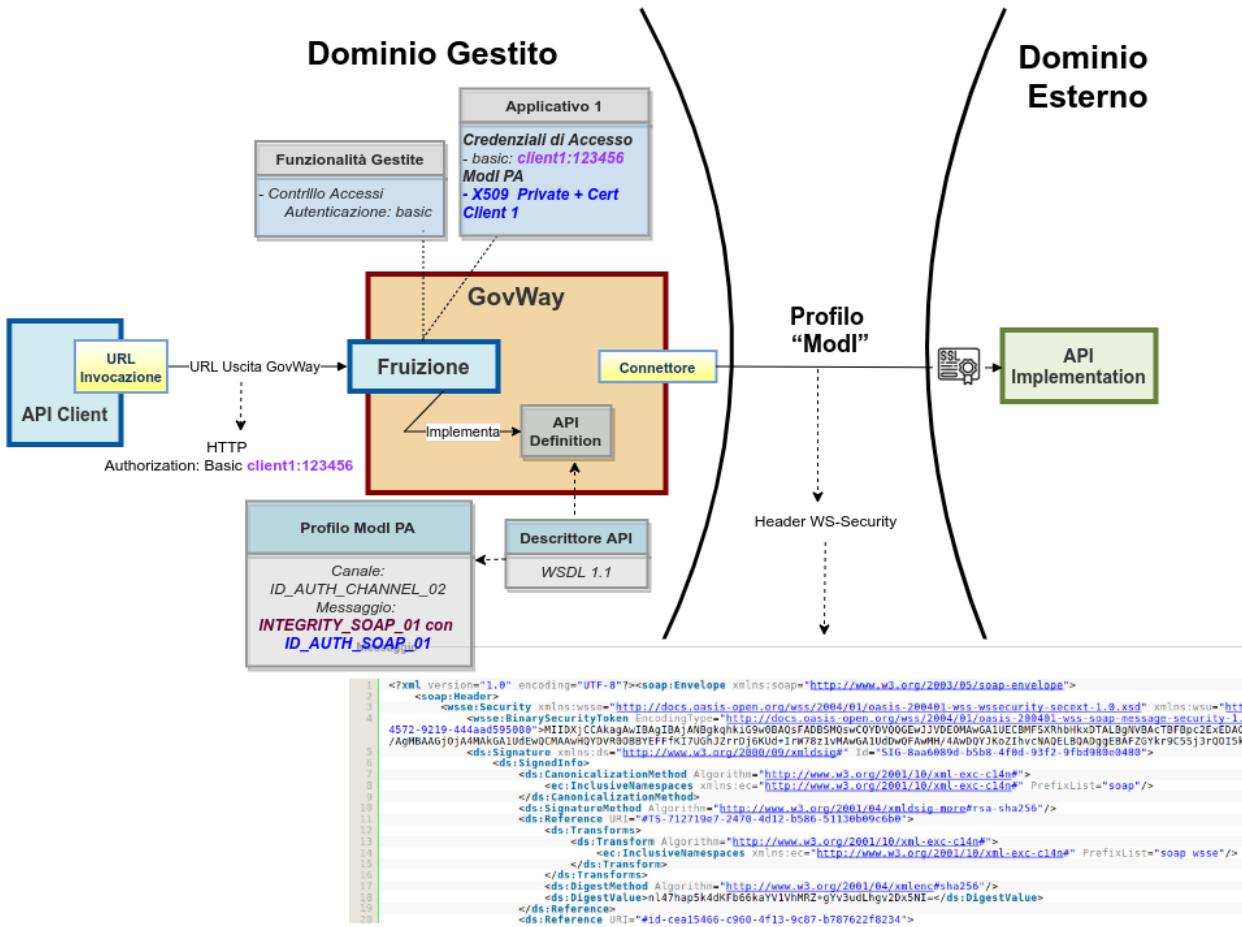


Fig. 3.78: Fruizione di una API SOAP con profilo “ModI”, pattern INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.79: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza utilizzato che in questo scenario è «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

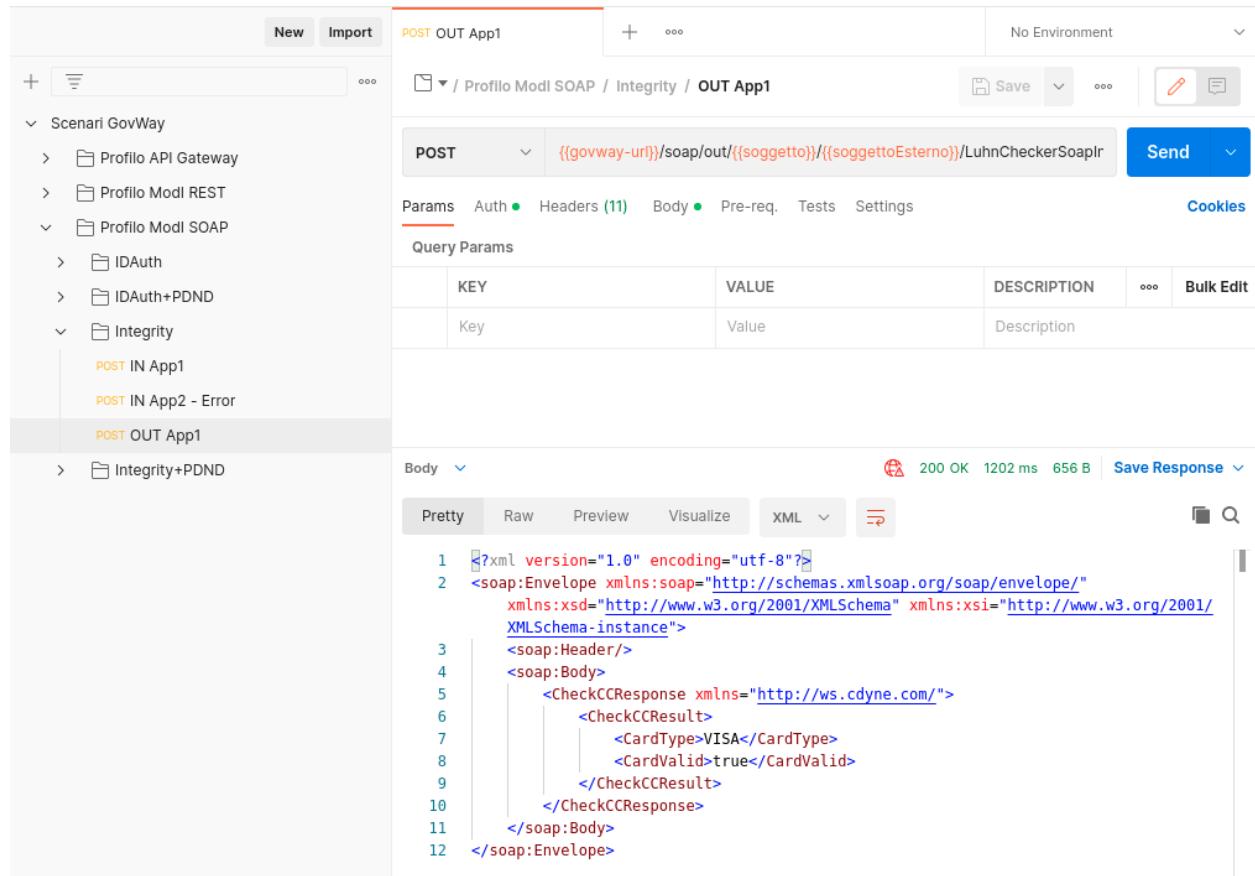


Fig. 3.80: Pattern Integrity - Esecuzione API SOAP, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in [Esecuzione](#).

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.81: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario [Configurazione](#) con le sole differenze dovuto al differente pattern di sicurezza utilizzato «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01».

Registrazione API

Viene registrata l’API «CreditCardVerificationIntegrity» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_02» (sicurezza canale) e «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.82).

Fruizione

Si registra la fruizione SOAP “LuhnCheckerSoapIntegrity”, relativa all’API precedentemente inserita, indicando i dati specifici nella sezione «ModI Richiesta» (Fig. 3.83).

La sezione «ModI Risposta» definisce i criteri per la validazione dei messaggi di risposta (Fig. 3.84).

3.3 Pattern “ID_AUTH” via PDND

Gli scenari riportati in questa sezione riguardano API configurate con pattern modipa_pdnd.

3.3.1 Erogazione API REST

Obiettivo

Esporre un servizio, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd.

Sintesi

Mostriamo in questa sezione come procedere per l’esposizione di un servizio REST da erogare nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo

API > CreditCardVerificationIntegrity v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

Modi

Sicurezza Canale

Pattern ▼

Direct Trust mutual Transport-Level Security

Sicurezza Messaggio

Pattern ▼

Integrità payload del messaggio

Applicabilità ▼

Digest Richiesta Non ripudiabilità della trasmissione (i)

Informazioni Utente Dati dell'utente che effettua la richiesta (i)

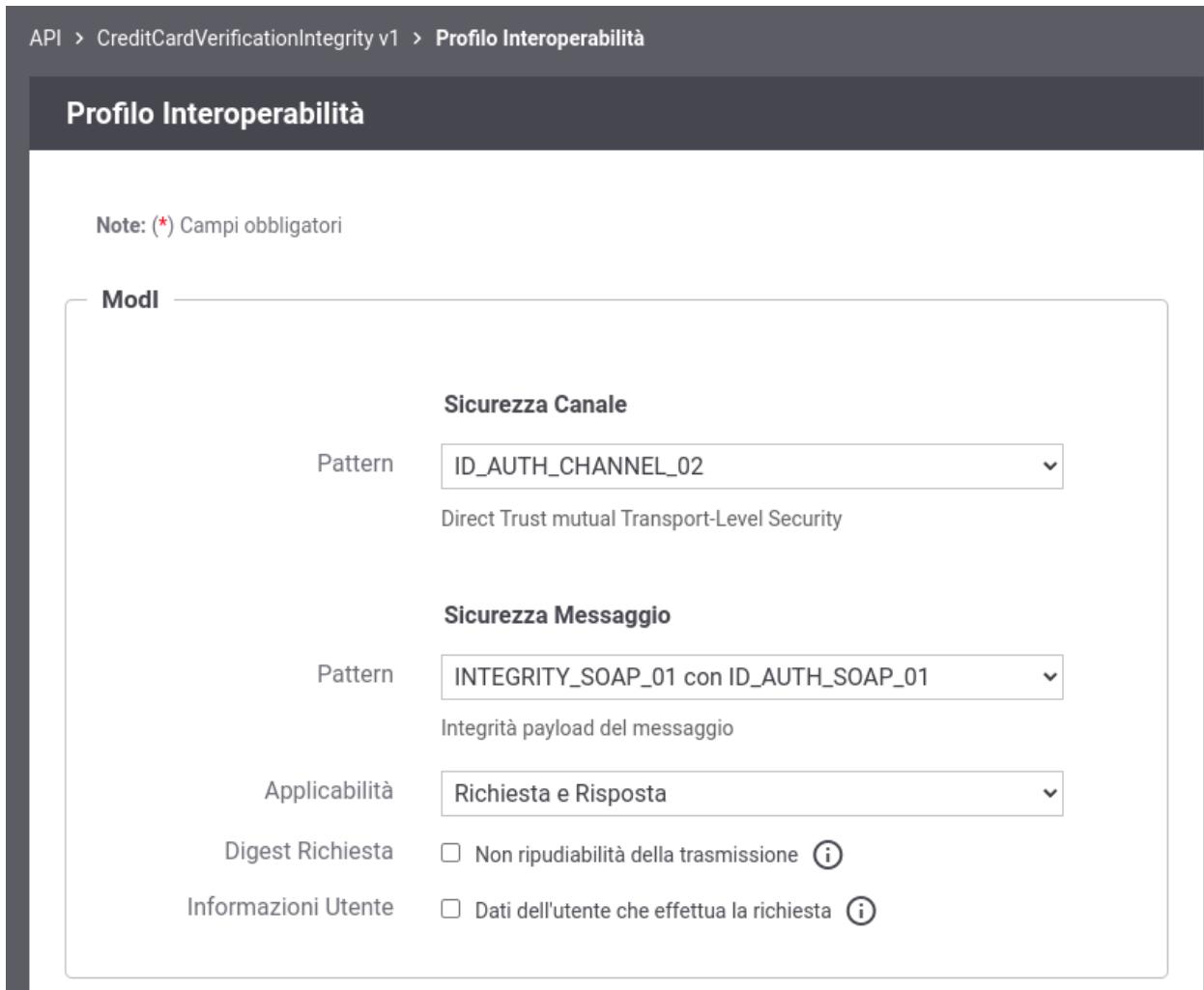


Fig. 3.82: Configurazione Pattern ModI «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» sulla API SOAP

Modi - Richiesta

Sicurezza Messaggio

Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token

WSAddressing To ⓘ

Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore

Fig. 3.83: Configurazione richiesta della fruizione

Modi - Risposta

Sicurezza Messaggio

TrustStore Certificati	Default
Time to Live	Default

Verifica WSAddressing To La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente

ⓘ

Fig. 3.84: Configurazione risposta della fruizione

ad illustrare lo scenario in cui un servizio è stato registrato sulla PDND, e i fruitori per poterlo fruire devono ottenere un voucher dalla PDND che successivamente devono inviare all'erogatore insieme alla normale richiesta di servizio.

La figura seguente descrive graficamente questo scenario.

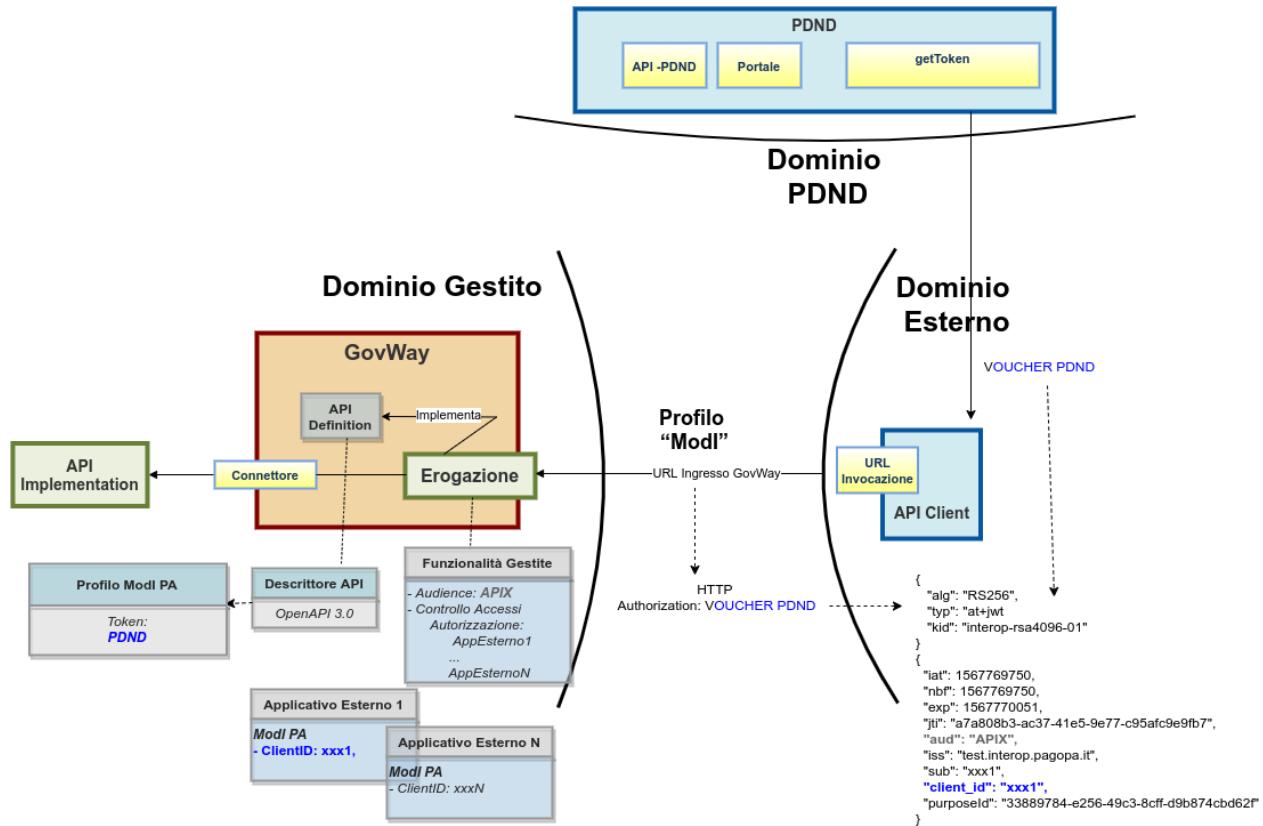


Fig. 3.85: Erogazione di una API REST con profilo “ModI”, pattern ID_AUTH_REST_01 via PDND

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, registrandolo sulla PDND
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND».

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menu in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.86: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API REST di esempio (PetStore) definita con pattern di interazione “CRUD” e pattern di sicurezza «ID_AUTH_CHANNEL_01» e «ID_AUTH_REST_01» via PDND»;
- un'istanza Govway per la gestione del profilo ModI nel dominio dell'erogatore;
- un'authorization server che simula la PDND;
- un client del dominio esterno che invoca la risorsa «POST /pet» dell'erogazione esposta da Govway;
- il server PetStore di esempio che riceve le richieste inoltrate dal Govway e produce le relative risposte. Per questo scenario viene utilizzato il server disponibile on line all'indirizzo “<https://petstore.swagger.io/>”.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth+PDND - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

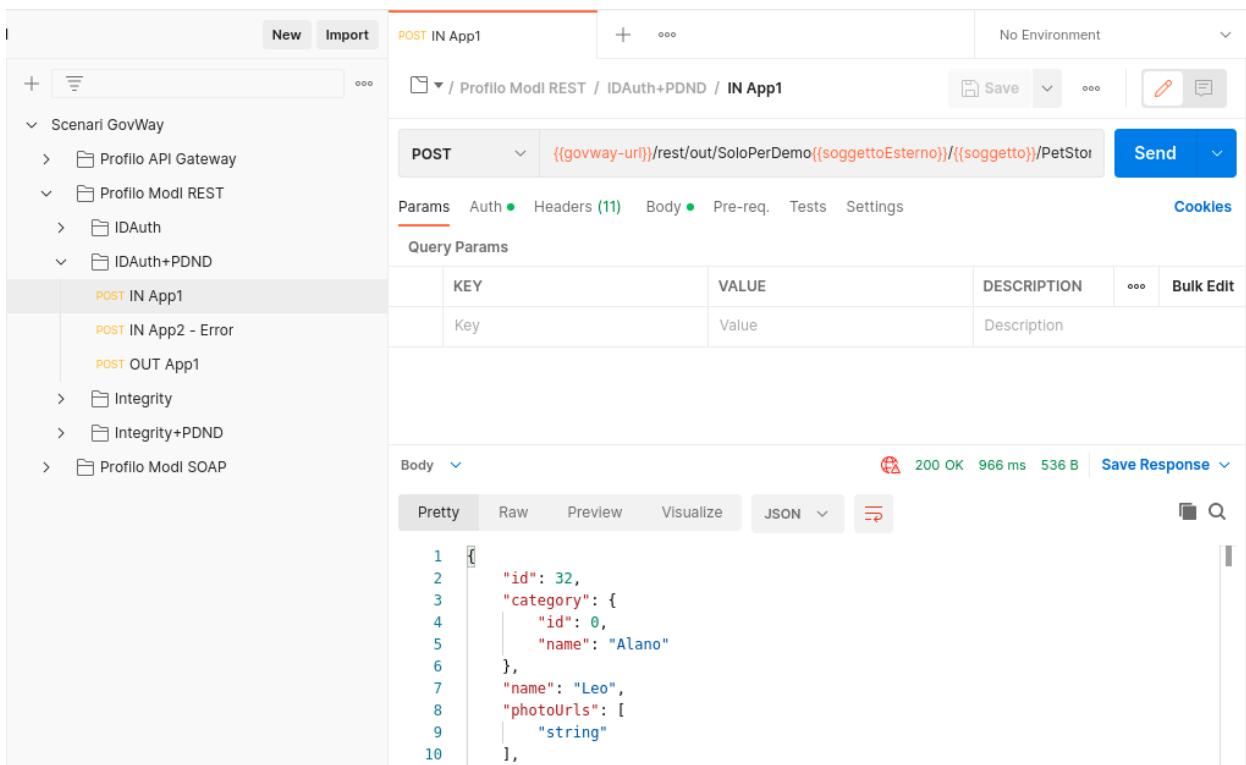


Fig. 3.87: Pattern IDAuth+PDND - Erogazione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

1. Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruttore, come in Fig. 3.88. Come si nota, al payload JSON è associato un insieme di header HTTP tra i quali «Authorization», che contiene il token di sicurezza che il fruttore ha ottenuto dalla PDND.

Headers	
Nome	
Content-Type	application/json
X-Message-Id	1f46c4b4-4f9b-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	cde738cd-acfc-4785-a59a-eb751595a001
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybmc8uZ292d2F5Lm9y h2UWZIHrQDLuBSuHsJQWfc2Wp16rbtLxvMqKSONk6lxmWknBch1hXBwzeTmPAkNHcDoYpqhmdR
X-Forwarded-Port	443
Pragma	no-cache
Accept-Encoding	gzip, deflate, br

Fig. 3.88: Messaggio inviato dal fruttore

2. Grazie alle configurazioni presenti nell'erogazione, ed in particolare all'indicazione che il token ricevuto deve essere validato tramite Token Policy PDND, GovWay è in grado di validare i dati di sicurezza ricevuti (Fig. 3.89) e decodificare il token.

2022-10-20 11:06:27.473	infolntegration	RicezioneBuste	Gestione Token [PDND] (Validazione JWT) in corso ...
2022-10-20 11:06:27.474	infolntegration	RicezioneBuste	Gestione Token [PDND] (Validazione JWT) completata con successo

Fig. 3.89: Evidenza diagnostica della validazione del token

3. Analizzando il token ricevuto nella sezione header (Fig. 3.90) si può notare che non viene riportata l'identità del fruttore tramite certificato X.509 come avveniva per il pattern ID_AUTH_REST_01 descritto nella scenario *Esecuzione*. L'identità del fruttore è presente nella sezione payload (Fig. 3.91) all'interno del claim *client_id*, insieme ai riferimenti temporali (iat, nbf, exp) e all'audience (aud). Da notare inoltre la presenza del claim "purposeId" che indica la finalità per cui il fruttore sta fruendo del servizio.

HEADER: ALGORITHM & TOKEN TYPE
<pre>{ "typ": "at+jwt", "alg": "RS256", "use": "sig", "kid": "interop-rsa4096-01" }</pre>

Fig. 3.90: Sezione «Header» del Token PDND

Nota: Il token ritornato dall'authorization server demo che simula la PDND contiene ulteriori claims che possono essere ignorati poiché non utili alla descrizione dello scenario e non presenti in un token PDND reale.

4. Il messaggio ricevuto dal Govway viene quindi validato, sulla base della configurazione realizzata, verificando in questo caso l'identità del fruttore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso. Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Se il processo di validazione del token ha successo è possibile consultare i dati interni al token ricevuto tramite la console come mostrato nelle figure Fig. 3.92 e Fig. 3.93.
5. Esaminando il messaggio inoltrato al backend è possibile vedere come tra gli header HTTP inoltrati vi sia l'header "GovWay-Token-PurposeId" contenente il valore del claim "purposeId" presente nel token ricevuto dalla PDND (Fig. 3.94).
6. Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo "App1-ModI" identificato grazie al claim "client_id" presente all'interno del token. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth+PDND - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

- La sicurezza messaggio applicata è quella dei pattern «ID_AUTH_REST_01 via PDND» come ampiamente mostrato precedentemente dove sono stati mostrati i token validati e i criteri autorizzativi.



```
PAYLOAD: DATA

{
  "aud": "PetStore",
  "sub": "App1-Esterno-PDND",
  "client_id": "App1-Esterno-PDND",
  "purposeId": "54806042-5e7f-4c70-9ee0-a4f100a079f7",
  "iss": "test.interop.pagopa.it",
  "exp": 1666258251,
  "iat": 1666257651,
  "nbf": 1666257651,
  "jti": "32c30a37-ed2e-4a50-a42e-0093b50773dc"
}
```

Fig. 3.91: Sezione «Payload» del Token PDND

2. L'identificazione del fruttore avviene rispetto al claim “client_id” presente all'interno del token. È stato anche mostrato come sia possibile configurare criteri autorizzativi puntuali.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.96: Profilo ModI della govwayConsole

Registrazione API

Viene registrata l'API «PetStoreAuthPDND» con il relativo descrittore OpenAPI 3. Viene selezionato il solo pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) mentre non deve essere selezionato alcun pattern di sicurezza messaggio nella sezione «ModI» poichè la gestione del token avverrà tramite validazione di un token OAuth attivato sull'erogazione (Fig. 3.97).

Applicativo Esterno

È opzionalmente possibile registrare l'applicativo esterno che corrisponde al fruttore del servizio. Questa scelta può essere fatta in base al tipo di autorizzazione che si è impostata sui fruttori. Vediamo i seguenti casi:

- Se si desidera autorizzare qualsiasi fruttore proveniente dalla PDND, questo passo può anche essere omesso. La validazione del token è sufficiente a stabilire che il fruttore ha ottenuto un voucher dalla PDND valido per il servizio invocato.

Transazioni > Ricerca Base > **Dettagli Transazione**

Dettagli Transazione

Informazioni Generali Informazioni Mittente Dettagli Messaggio Diagnostici Informazioni Avanzate

Informazioni Mittente

Fruitore EnteEsterno
Applicativo Fruitore App1-PDND
ID Autenticato /o=govway.org/c=it/cn=enteEsterno.govway.org/
Metodo HTTP POST
URL Invocazione [in] /govway/rest/in/Ente/PetStoreAuthPDND/v1/pet
Client IP 172.20.0.2
X-Forwarded-For 172.20.0.2
Codice Risposta Client 200

Token

Issuer https://govway.localdomain/auth/realm/master
Subject 3210f474-773c-44f6-a25b-8999c796f7c7
Client ID App1-Esterno-PDND
Applicativo Client App1-PDND
Token [Visualizza](#)

Fig. 3.92: Dati principali presenti nel Token PDND

Transazioni > Ricerca Base > Dettagli Transazione > Token

Token

```
1  {
2      "type" : "validated_token",
3      "valid" : true,
4      "iss" : "https://govway.localdomain/auth/realm/master",
5      "sub" : "3210f474-773c-44f6-a25b-8999c796f7c7",
6      "aud" : [ "PetStore", "CreditCardVerification", "account" ],
7      "exp" : 1666256847000,
8      "iat" : 1666256787000,
9      "clientId" : "App1-Esterno-PDND",
10     "jti" : "f123ccee-f513-472a-bac3-af2c59c64285",
11     "scopes" : [ "email", "profile" ],
12     "userInfo" : { },
13     "claims" : {
14         "sub" : "3210f474-773c-44f6-a25b-8999c796f7c7",
15         "email_verified" : "false",
16         "clientHost" : "172.20.0.2",
17         "iss" : "https://govway.localdomain/auth/realm/master",
18         "purposeId" : "b149ca3c-4edf-11ed-80f4-0242ac140002",
19         "typ" : "Bearer",
20         "preferred_username" : "service-account-app1-esterno-pdnd",
21         "clientAddress" : "172.20.0.2",
22         "client_id" : "App1-Esterno-PDND",
```

Fig. 3.93: Claim presenti nel Token PDND

Headers	
Nome	Valore
X-Real-Ip	172.20.0.1
GovWay-Token-ClientId	App1-Esterno-PDND
GovWay-Token-Audience	PetStore,CreditCardVerification,account
GovWay-Sender	EnteEsterno
Cache-Control	no-cache
GovWay-Application	App1-PDND
GovWay-Token-Jti	51bb4e16-1592-43a4-a263-070ed8a58241
GovWay-Token-Issuer	https://govway.localdomain/auth/realms/master
GovWay-Transaction-ID	cba1b693-5072-11ed-a5ac-0242ac140002
Content-Type	application/json
GovWay-Token-PurposeId	b149ca3c-4edf-11ed-80f4-0242ac140002
User-Agent	GovWay
GovWay-Token-Application	App1-PDND

Fig. 3.94: Header HTTP “GovWay-Token-PurposeId” inoltrato al backend

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Shows a tree structure of API profiles: "Scenari GovWay" (selected), "Profilo API Gateway", "Profilo Modl REST" (selected), "IDAAuth", "IDAAuth+PDND" (selected), "POST IN App1", "POST IN App2 - Error" (selected), "POST OUT App1", "Integrity", "Integrity+PDND", and "Profilo Modl SOAP".
- Header Bar:** Shows "New", "Import", "POST IN App2 - Error", "No Environment", and a dropdown for "Profile Modl REST / IDAuth+PDND / IN App2 - Error".
- Request Section:**
 - Method: POST
 - URL: `({govway-url})/rest/out/SoloPerDemo({soggettoEsterno})/({soggetto})/PetStore`
 - Headers: Content-Type: application/json, Authorization: Bearer b149ca3c-4edf-11ed-80f4-0242ac140002
 - Body: JSON response showing a 403 Forbidden error with the following content:

```

1  [
2    {
3      "type": "https://govway.org/handling-errors/403/Authorization.html",
4      "title": "Authorization",
5      "status": 403,
6      "detail": "Authorization failed",
7      "govway_id": "7cffa20e-505a-11ed-a5ac-0242ac140002"
8    }
9  ]

```
- Bottom Status Bar:** Shows "403 Forbidden 78 ms 446 B" and "Save Response".

Fig. 3.95: Pattern IDAuth+PDND - Erogazione API REST - Autorizzazione negata, esecuzione da Postman

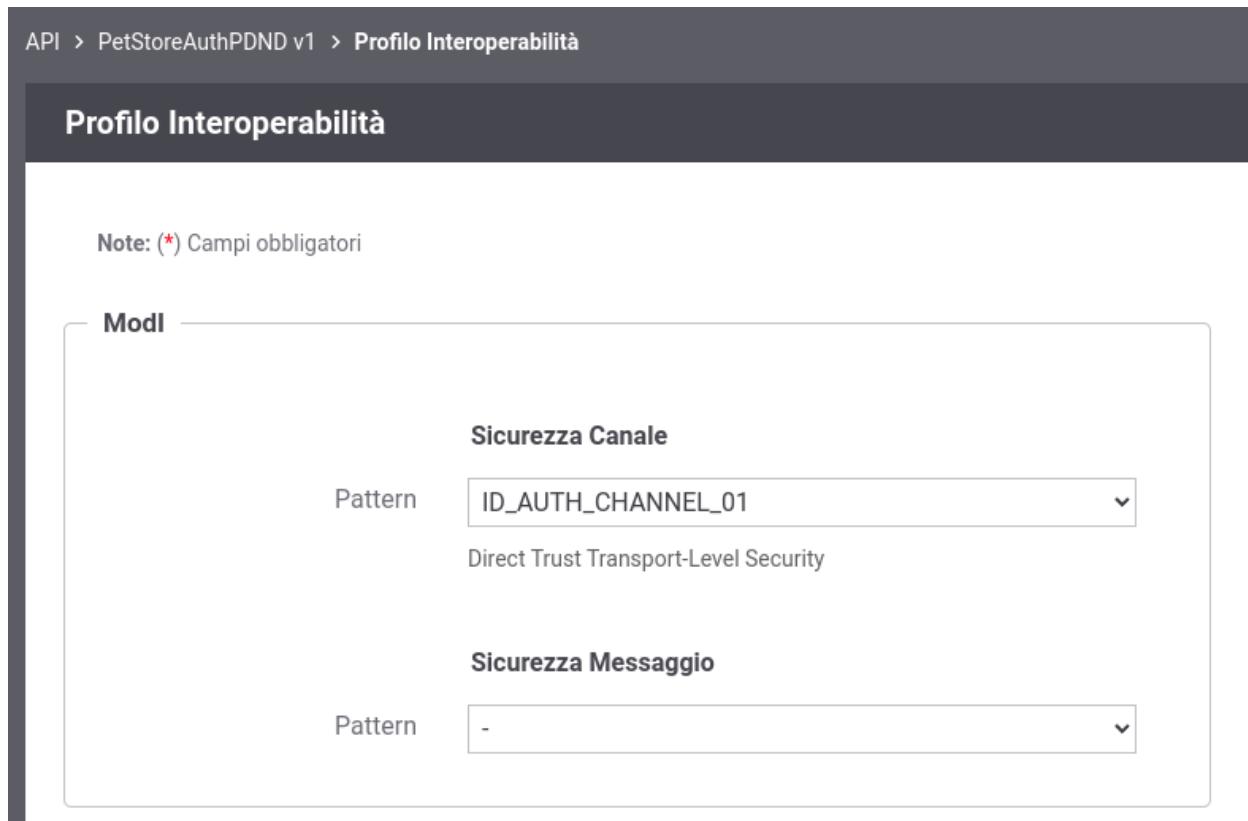


Fig. 3.97: Configurazione Pattern ModI con «ID_AUTH_CHANNEL_01» senza sicurezza messaggio

- In alternativa è possibile configurare una autorizzazione puntuale procedendo alla registrazione degli applicativi fornendo i singoli “client_id” necessari all’identificazione (Fig. 3.98). Questo scenario è quello preconfigurato.

Applicativo

Profilo Interoperabilità	Modi
Dominio	Esterno
Soggetto	EnteEsterno
Nome *	App1-PDND
Tipo	Client
Proprietà(0)	

Ruoli

visualizza(0)

Modi

Sicurezza Messaggio	Authorization PDND
ClientId registrato sulla PDND	
Token Policy *	PDND
Identificativo *	App1-Esterno-PDND

Fig. 3.98: Configurazione applicativo esterno (fruitore)

Token Policy PDND

Con il prodotto viene fornita built-in la token policy “PDND” (Fig. 3.99) da finalizzare nella sezione “TrustStore” nei seguenti aspetti (Fig. 3.100):

- File: deve essere indicato un path su file system che contiene il certificato di firma della PDND ottenibile tramite la url “`.../.well-known/jwks.json`” fornita dalla PDND stessa;
- Alias Certificato: deve contenere l’alias (il kid) della chiave pubblica utilizzata dalla PDND per firmare i token rilasciati, corrispondente al valore del claim “`kid`” presente nel JWKSet configurato al punto precedente;
- Token Forward: deve essere eventualmente configurata la modalità di forward delle informazioni presenti nel token verso il backend, utile nel nostro scenario per far arrivare il valore del claim “`purposeId`” al backend nell’header HTTP “`GovWay-Token-PurposeId`”.

Erogazione

Si registra l’erogazione «`PetStoreAuthPDND`», relativa all’API precedentemente inserita, abilitando la validazione del token ricevuto dalla PDND tramite la omonima policy (Fig. 3.101).

Token Policy > PDND

PDND

Note: (*) Campi obbligatori

Token Policy

Tipo	Validazione
Nome	PDND
Descrizione	<input type="text"/>

Informazioni Generali

Token

Tipo	<input type="text" value="JWS"/>
Posizione	<input type="text" value="RFC 6750 - Bearer Token Usage"/>

Elaborazione Token

Validazione JWT	<input checked="" type="checkbox"/>
Token Introspection	<input type="checkbox"/>
OIDC - UserInfo	<input type="checkbox"/>
Token Forward	<input checked="" type="checkbox"/>

Fig. 3.99: Token Policy PDND (Dati Generali)

Fig. 3.100: Token Policy PDND (Aspetti da Configurare)

Si può notare nella sezione “Autenticazione Canale” del Controllo degli Accessi come l’autenticazione https sia opzionale per essere aderenti al pattern di sicurezza canale «ID_AUTH_CHANNEL_01» (Fig. 3.102).

Nella sezione “Autorizzazione” si può invece vedere come nella voce “Autorizzazione per Token Claims” vi sia configurato il valore del claim “aud” atteso.

Se si è scelto inoltre di registrare gli applicativi esterni, fruitori del servizio, saranno specificati i singoli applicativi fruitori autorizzati ad effettuare richieste al servizio erogato. Questo scenario è quello preconfigurato come mostrato nelle figure Fig. 3.103 e Fig. 3.104.

3.3.2 Fruizione API REST

Obiettivo

Fruire di un servizio REST, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd.

Sintesi

Mostriamo in questa sezione come procedere per l’integrazione di un applicativo con un servizio REST erogato nel rispetto della normativa italiana alla base dell’interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui il servizio è stato registrato sulla PDND, e il fruitore per poterlo fruire deve ottenere un voucher dalla PDND che successivamente deve inviare all’erogatore insieme alla normale richiesta di servizio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

Erogazioni > PetStoreAuthPDND v1 (Ente) > Configurazione > **Controllo Accessi**

Controllo Accessi

Note: (*) Campi obbligatori

^ Autenticazione Token

Stato	abilitato
Policy *	PDND
Validazione JWT	abilitato
Token Forward	abilitato

Required Claims

Issuer	<input type="checkbox"/>
ClientId	<input checked="" type="checkbox"/>
Subject	<input type="checkbox"/>
Username	<input type="checkbox"/>
eMail	<input type="checkbox"/>

Fig. 3.101: Controllo degli Accessi - Autenticazione Token

^ Autenticazione Canale

Stato	https
Opzionale	<input checked="" type="checkbox"/>

Fig. 3.102: Controllo degli Accessi - Autenticazione Canale

Autorizzazione

Stato

Autorizzazione Canale

per Richiedente

per Ruoli

Autorizzazione Messaggio

per Richiedente

Applicativi (1)

per Ruoli

Autorizzazione per Token Claims

Abilitato

Claims

Indicare per riga i claims richiesti (nome=valore); visualizzare 'info' per maggiori dettagli

Fig. 3.103: Controllo accessi con autorizzazione dell'audience e degli applicativi esterni

Erogazioni > PetStoreAuthPDND v1 (Ente) > Configurazione > Controllo Accessi > **Autorizzazione Messaggio - Applicativi**

Autorizzazione Messaggio - Applicativi

Visualizzati record [1-1] su 1

<input type="checkbox"/>	Soggetto	Applicativo	<input type="checkbox"/>
<input type="checkbox"/>	EnteEsterno	App1-PDND	<input type="checkbox"/>

Fig. 3.104: Lista degli applicativi esterni autorizzati

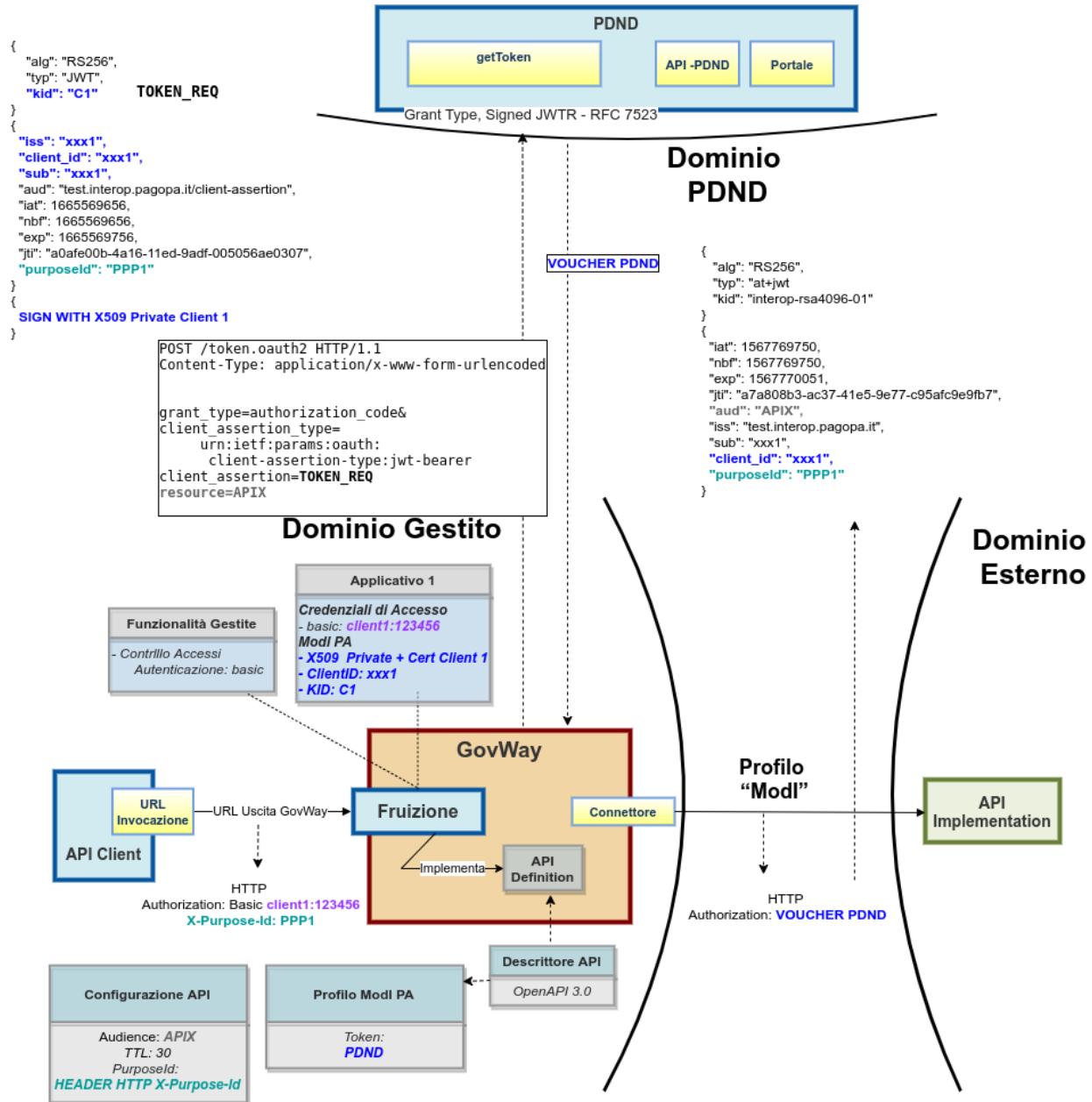


Fig. 3.105: Fruizione di una API REST con profilo "ModI", pattern ID_AUTH_REST_01 via PDND

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa registrata su PDND
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.106: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API REST di esempio (PetStore) definita con pattern di interazione “CRUD” e pattern di sicurezza «ID_AUTH_CHANNEL_01» e «ID_AUTH_REST_01 via PDND»;
- un'istanza Govway per la gestione del profilo ModI nel dominio del fruitore;
- un'authorization server che simula la PDND;
- un client che invoca la risorsa «POST /pet» con un messaggio di esempio diretto alla fruizione configurata su Govway.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - IDAuth+PDND - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto nelle diverse fasi dell'esecuzione andando a consultare la console “govwayMonitor”.

1. Il messaggio di richiesta inviato dal fruitore viene elaborato da Govway che, tramite la configurazione della firma digitale associata all'applicativo mittente, è in grado di produrre un token di sicurezza da inviare alla PDND con il quale ottenere indietro un voucher spendibile per il servizio desiderato. Tramite la console è possibile esaminare sia l'asserzione JWT inviata alla PDND (Fig. 3.108) che l'access token ottenuto dalla PDND (Fig. 3.109).
2. Esaminando l'header e il payload dell'asserzione JWT inviata alla PDND (Fig. 3.110) si può notare:
 - Valore del claim “kid” associato all'applicativo mittente in configurazione
 - Valore del claim “client_id” (uguale per i claim “sub” e “iss”) associato all'applicativo mittente in configurazione
 - Valore del claim “purposeId” indicato dal client (nell'esempio Postman) tramite un header http “X-Purpose-Id”
3. Analizzando l'access token ricevuto dalla PDND, nella sezione header (Fig. 3.111) si può notare che non viene riportata l'identità del fruitore tramite certificato X.509 come avveniva per il pattern ID_AUTH_REST_01 descritto nella scenario *Esecuzione*. L'identità del fruitore è presente nella sezione payload (Fig. 3.112) all'interno del claim *client_id*, insieme ai riferimenti temporali (iat, nbf, exp) e all'audience (aud) del servizio per cui si è

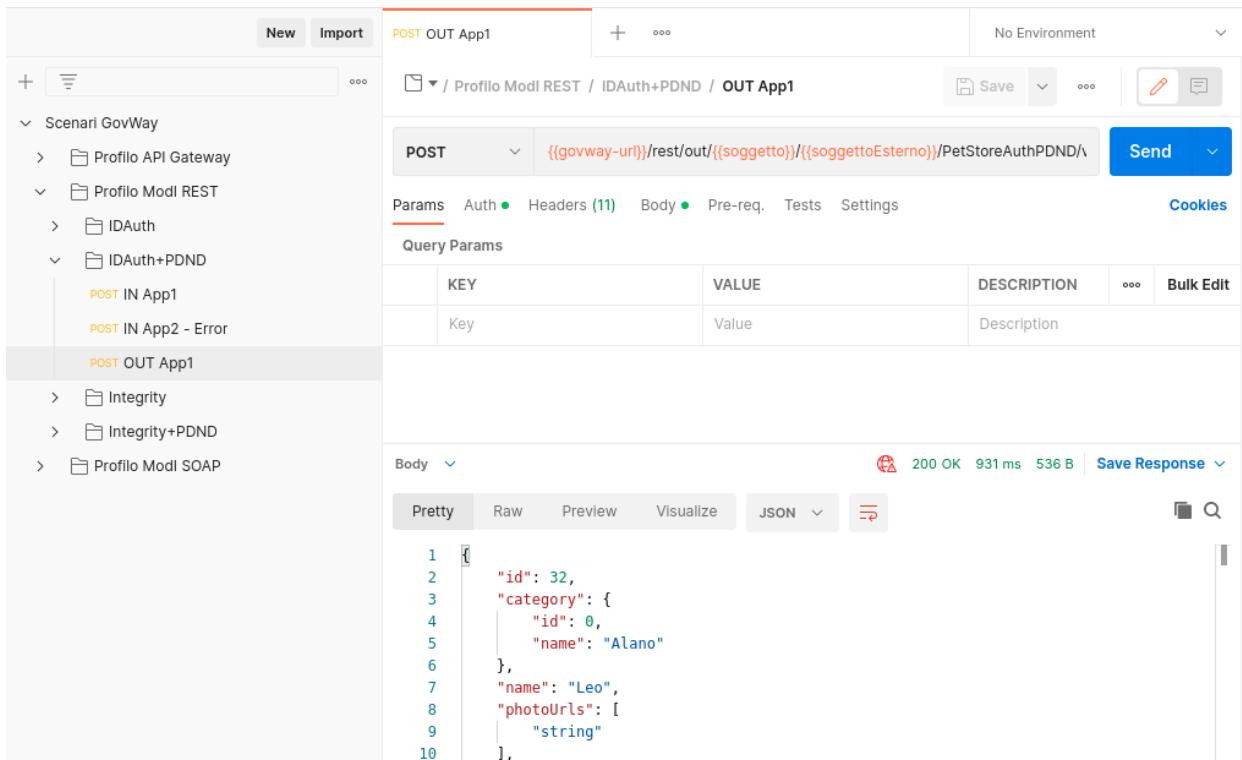


Fig. 3.107: Pattern IDAuth+PDND - Fruizione API REST, esecuzione da Postman

```

1  {
2    "type" : "retrieved_token",
3    "request" : {
4      "policy" : "KeyCloak-NegoziazionePDND",
5      "transactionId" : "1664c8e8-5076-11ed-a5ac-0242ac140002",
6      "grantType" : "rfc7523_x509",
7      "jwtClientAssertion" : [
8        "token" : "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVV0NHTzVac0VxeVBXenpxZ3RURkNYV1Bw",
9      ],
10     "endpoint" : "https://govway.localdomain/auth/realms/master/protocol/openid-connect/token",
11     "prepareRequest" : 1666270363102,
12     "sendRequest" : 1666270363108,
13     "receiveResponse" : 1666270363115,
14     "parseResponse" : 1666270363115,
15     "processComplete" : 1666270363115
16   },
17   "valid" : true,
18   "accessToken" : "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVV0NHTzVac0VxeVBXenpxZ3RURkNYV1Bw",
19   "refreshToken" : "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJiMmI2ODI5NC00Yjc0LTQ4NmQtODc0NS0",
20   "retrievedIn" : 1666270363115,
21   "expiresIn" : 1666270423115,
22   "retrievedRefreshTokenIn" : 1666270363115,

```

Fig. 3.108: Evidenza dell'asserzione JWT inviata alla PDND

Transazioni > Ricerca Base > Dettagli Transazione > **Token**

Token

```

1  {
2    "type" : "retrieved_token",
3    "request" : {
4      "policy" : "Keycloak-NegoziazionePDND",
5      "transactionId" : "1664c8e8-5076-11ed-a5ac-0242ac140002",
6      "grantType" : "rfc7523_x509",
7      "jwtClientAssertion" : {
8        "token" : "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6InpnQzZKbGNkanpkWmt3LXo2YVNxbHRwS2JZNWdncU1Ub3p3aFFjN0i
9      },
10     "endpoint" : "https://govway.localdomain/auth/realm/master/protocol/openid-connect/token",
11     "prepareRequest" : 1666270363102,
12     "sendRequest" : 1666270363108,
13     "receiveResponse" : 1666270363115,
14     "parseResponse" : 1666270363115,
15     "processComplete" : 1666270363115
16   },
17   "valid" : true,
18   "accessToken" : "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6InpnQzZKbGNkanpkWmt3LXo2YVNxbHRwS2JZNWdncU1Ub3p3aFFjN0i
19   "refreshToken" : "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6InpnQzZKbGNkanpkWmt3LXo2YVNxbHRwS2JZNWdncU1Ub3p3aFFjN0i
20   "retrievedIn" : 1666270363115,
21   "expiresIn" : 1666270423115,
22   "retrievedRefreshTokenIn" : 1666270363115,

```

Fig. 3.109: Evidenza dell'access token ottenuto dalla PDND

HEADER: ALGORITHM & TOKEN TYPE	<pre> { "alg": "RS256", "typ": "JWT", "kid": "zgC6JlcdjzdZkw-z6aSWltpKbY5ggqMTozwhQc7FU5M" } </pre>
PAYOUT: DATA	<pre> { "iss": "App1-PDND", "client_id": "App1-PDND", "sub": "App1-PDND", "aud": "https://govway.localdomain/auth/realm/master", "iat": 1666270363, "nbf": 1666270363, "exp": 1666270663, "jti": "1664c8e8-5076-11ed-a5ac-0242ac140002", "purposeId": "b149ca3c-4edf-11ed-80f4-0242ac140002" } </pre>

Fig. 3.110: Header e Payload dell'asserzione JWT inviata alla PDND

richiesto il voucher. Da notare inoltre la presenza del claim “purposeId” che servirà ad indicare la finalità per cui il fruitore sta fruendo del servizio all’erogatore.

```
HEADER: ALGORITHM & TOKEN TYPE

{
  "typ": "at+jwt",
  "alg": "RS256",
  "use": "sig",
  "kid": "interop-rsa4096-01"
}
```

Fig. 3.111: Sezione «Header» del Token PDND

```
PAYLOAD: DATA

{
  "aud": "PetStore",
  "sub": "App1-Esterno-PDND",
  "client_id": "App1-Esterno-PDND",
  "purposeId": "54806042-5e7f-4c70-9ee0-a4f100a079f7",
  "iss": "test.interop.pagopa.it",
  "exp": 1666258251,
  "iat": 1666257651,
  "nbf": 1666257651,
  "jti": "32c30a37-ed2e-4a50-a42e-0093b50773dc"
}
```

Fig. 3.112: Sezione «Payload» del Token PDND

Nota: Il token ritornato dall’authorization server demo che simula la PDND contiene ulteriori claims che possono essere ignorati poichè non utili alla descrizione dello scenario e non presenti in un token PDND reale.

4. Tramite la console govwayMonitor si può visualizzare il messaggio di richiesta in uscita che è il medesimo di quello in entrata con la differenza che è stato aggiunto l’access token ottenuto dalla PDND tra gli header HTTP all’interno dell’header «Authorization» (Fig. 3.113).
5. Govway riceve la risposta dell’erogatore grazie al fatto che ha inviato un voucher PDND correttamente validato dall’erogatore.

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. Viene effettuata una negoziazione del voucher PDND come mostrato nelle tracce relative ai token scambiati con la PDND.

Headers	
Nome	
Content-Type	application/json
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
X-Forwarded-Port	443
Accept-Encoding	gzip, deflate, br
Postman-Token	d924391e-10cd-4c75-8063-4cbfaa74639a
User-Agent	GovWay
Accept	/*
GovWay-Message-ID	5ade2322-4fac-11ed-a5ac-0242ac140002
GovWay-Transaction-ID	5acd8134-4fac-11ed-a5ac-0242ac140002
Authorization	Bearer eyJhbGciOiJSUzI1NilsInR5cCI6IkpXVCIsImtpZCI6ImFwcDEuZW50ZS5nb3Z3YXkub3JnWylSJxWAFBE4zpeb4JpJRwmafmwJLqddHy7j8bMjGx9x3lGOws6AhiTAKaK2HPGbpD

Fig. 3.113: Messaggio di richiesta in uscita (con voucher PDND inserito nell'header HTTP)

2. L'invocazione del servizio avviene fornendo il voucher della PDND precedentemente negoziato.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.114: Profilo ModI della govwayConsole

Registrazione API

Viene registrata l'API «PetStoreAuthPDND» con il relativo descrittore OpenAPI 3. Viene selezionato il solo pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) mentre non deve essere selezionato alcun pattern di sicurezza messaggio nella sezione «ModI» poichè la gestione del token avverrà tramite validazione di un token OAuth attivato sull'erogazione (Fig. 3.115).

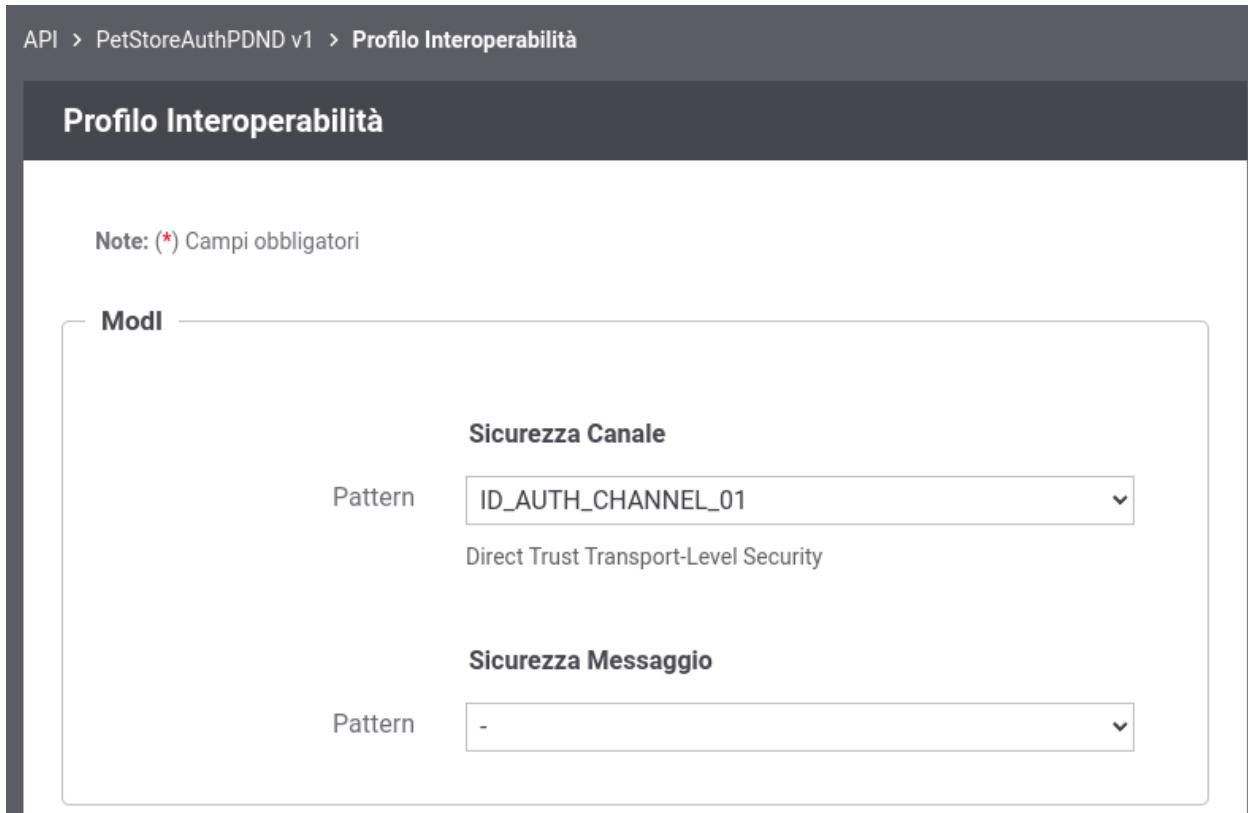


Fig. 3.115: Configurazione Pattern ModI con «ID_AUTH_CHANNEL_01» senza sicurezza messaggio

Applicativo

Si configura l'applicativo mittente indicando, nella sezione ModI, i parametri del keystore e i parametri di identificazione sulla PDND necessari affinché Govway possa produrre il token di sicurezza firmando per conto dell'applicativo (Fig. 3.116 e Fig. 3.117). Alla registrazione dell'applicativo vengono associate credenziali “basic” che consentono a GovWay di identificarlo.

Applicativo	
Dominio	Interno
Soggetto	Ente
Nome *	App1-PDND
Tipo	Client
<u>Proprietà(0)</u>	

Modalità di Accesso	
Tipo	http-basic
Utente *	App1-PDND.ente
Modifica Password	<input type="checkbox"/>

Fig. 3.116: Configurazione applicativo fruitore (Dati Generali)

Token Policy PDND

Per la configurazione delle fruizioni con un pattern di sicurezza via PDND è necessario registrare una Token Policy di Negoziazione del tipo descritto nella sezione “tokenNegoziazionePolicy_pdnd”.

Una volta effettuata la registrazione della Token Policy, per utilizzarla in una fruizione è sufficiente associarla al connettore della fruizione come descritto nella sezione avanzate_connatori_tokenPolicy.

Di seguito vengono riportati tutte le informazioni più importanti della policy:

- Tipo: SignedJWT;
- PDND: flag attivato;
- URL: endpoint esposto dalla PDND su cui è possibile richiedere lo stacco del voucher;
- JWT Keystore: parametri di accesso al keystore contenente la chiave privata corrispondente al certificato X509 caricato sulla PDND durante la registrazione dell'applicativo client. I parametri variano in funzione del tipo di keystore selezionato e nello scenario preconfigurato è stata scelta la modalità “Definito nell'applicativo ModI” nella quale il keystore utilizzato per firmare l'asserzione JWT inviata alla PDND sarà quello definito nell'applicativo ModI richiedente (Fig. 3.119).

Nota: Questa modalità consente di definire una unica TokenPolicy di negoziazione utilizzabile da più applicativi richiedenti ognuno configurato con la propria coppia di chiavi di firma e i relativi identificativi “client_id” e “kid”.

Modi - Sicurezza Messaggio

KeyStore

Abilitato

Modalità

Path *****

Tipo

Password *****

Alias Chiave Privata *****

Password Chiave Privata *****

Certificato No file chosen

Authorization ModI

Identificativo Client

Identificativo dell'Applicativo scambiato nei token di sicurezza

URL (x5u)

URL che riferisce un certificato (o certificate chain) X.509 corrispondente alla chiave firmataria del security token

Authorization OAuth

Abilitato

Token Policy di Validazione

!Attenzione!! Per consentire un'identificazione dell'applicativo su API erogate da altri soggetti di dominio interno selezionare una token policy.

Identificativo *****

Key Id (kid) del Certificato

Fig. 3.117: Configurazione applicativo fruitore (Configurazione ModI)

Token Policy > KeyCloak-NegoziazionePDND

KeyCloak-NegoziazionePDND

Note: (*) Campi obbligatori

Token Policy

Tipo	Negoziazione
Nome	KeyCloak-NegoziazionePDND
Descrizione	<input type="text"/>

Token Endpoint

Tipo	Signed JWT
PDND	<input checked="" type="checkbox"/>
URL *	<input type="text" value="https://govway.localdomain/auth/realm..."/> <small>i</small>
Connection Timeout *	<input type="text" value="5000"/>
Read Timeout *	<input type="text" value="10000"/>
Https	<input checked="" type="checkbox"/>
Proxy	<input type="checkbox"/>

Fig. 3.118: Token Policy di Negoziazione PDND (Endpoint)

- JWT Signature: algoritmo di firma
- JWT Header:
 - Type (typ): lasciare il valore “JWT”;
 - Key Id (kid): deve essere indicato l’identificativo univoco (KID) associato al certificato caricato sulla PDND e ottenuto al termine della registrazione dell’applicativo client. Può essere fornito tramite differenti modalità e nello scenario preconfigurato è stata scelta la modalità “Definito nell’applicativo ModI” nella quale il valore del KID viene configurato sull’applicativo richiedente (Fig. 3.119).

JWT KeyStore

Tipo Definito nell'applicativo ModI

JWT Signature

Signature Algorithm RS256

JWT Header

Key Id (kid) Definito nell'applicativo ModI

X.509 Certificate -

Digest X.509 Certificate -

Type (typ) * JWT

Content Type (cty)

Fig. 3.119: Token Policy di Negoziazione PDND (Keystore definito nell’applicativo ModI)

- JWT Payload:

l’identificativo univoco dell’applicativo client (“*client_id*” o “*sub*”) ottenuto al termine della registrazione dell’applicativo sulla PDND deve essere indicato nei seguenti campi:

- Client ID
- Issuer
- Subject

Nello scenario preconfigurato è stato però scelta la modalità alternativa in cui il ClientID ottenuto dalla PDND deve essere configurato sull’applicativo richiedente e la token policy viene configurata per utilizzare tale valore (Fig. 3.120).

Gli altri campi presenti nella sezione “JWT Payload” rappresentano (Fig. 3.120):

- Audience: indica il servizio di stacco del voucher della PDND. Il valore, fornito dalla PDND, è indipendente dal servizio per cui si vuole richiedere un voucher e varia solamente in funzione dell’ambiente di validazione o produzione della PDND stessa;

- Identifier: consente di configurare la modalità di valorizzazione del claim “jti” presente all’interno del token di richiesta inviato alla PDND. Si suggerisce di valorizzare il campo con la keyword “\${transaction:id}” al fine di utilizzare l’identificativo di transazione della richiesta;
- Time to Live (secondi): consente di indicare la durata del token di richiesta inviato alla PDND (es. 100 sec);
- Purpose ID: identificativo univoco della finalità per cui si intende fruire di un servizio. Il valore può essere fornito staticamente o può contenere una keyword risolta a runtime in modo da valorizzare il claim purposeId con un valore prelevato dai dati della richiesta o dalla configurazione della fruizione. Nello scenario preconfigurato il purposeId viene indicato dall’applicativo richiedente tramite l’header HTTP “X-Purpose-Id”.
- Informazioni Sessione: consente di valorizzare il claim “sessionInfo” previsto dalla PDND. La valorizzazione può essere statica o formata da parti dinamiche risolte a runtime dal Gateway (per maggiori dettagli valoriDinamici).

JWT Payload

Client ID	Definito nell’applicativo Modl	▼
Issuer	ClientID dell’applicativo Modl	▼
Subject	ClientID dell’applicativo Modl	▼
Audience *	https://govway.localdomain/auth/realm/master	ⓘ
Identifier	\${transaction:id}	ⓘ
Time to Live (secondi) *	300	
Indica la validità temporale, in secondi, a partire dalla data di creazione dell’asserzione		
Purpose ID *	\${header:X-Purpose-Id}	ⓘ
Informazioni Sessione	<div style="border: 1px solid #ccc; height: 100px; margin-bottom: 10px;"></div> <p>Indicare per riga i claims (nome=valore) da aggiungere nell’oggetto ‘sessionInfo’</p>	
Claims	<div style="border: 1px solid #ccc; height: 100px;"></div> <p>Indicare per riga gli ulteriori claims (nome=valore)</p>	

Fig. 3.120: Token Policy di Negoziazione PDND (JWT Payload)

- Dati Richiesta:
 - Resource: indica l’audience/url del servizio per cui si vuole richiedere un voucher; nello scenario preconfigurato il valore viene preso dalla proprietà “PDND-resource” della fruizione configurata.
 - Client ID: deve essere indicato il medesimo valore inserito nel campo “Client ID” della sezione “JWT Payload”; nello scenario preconfigurato viene infatti utilizzato il valore configurato sull’applicativo richiedente.

Dati Richiesta

Scope	<input type="text"/>	
Elencare più scope separandoli con la virgola		
Audience	<input type="text"/>	
Client ID	ClientID dell'applicativo Modl	
Resource	<input type="text" value="\${config:PDND-resource}"/>	
Parametri	<input type="text"/>	
Indicare per riga gli ulteriori parametri (nome=valore)		

Fig. 3.121: Token Policy di Negoziazione PDND (Dati Richiesta)

Fruizione

Si registra la fruizione «PetStoreAuthPDND», relativa all'API precedentemente inserita, indicando l'utilizzo della token policy di negoziazione sul connettore (Fig. 3.122).

Tra le proprietà della fruizione viene definita la proprietà “PDND-resource” contenente il valore da inserire nella richiesta di voucher effettuata alla PDND che identifica il servizio per cui si sta richiedendo il token (Fig. 3.123).

3.3.3 Erogazione API SOAP

Obiettivo

Esporre un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd.

Sintesi

Mostriamo in questa sezione come procedere per l'esposizione di un servizio SOAP da erogare nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui un servizio è stato registrato sulla PDND, e i fruitori per poterlo fruire devono ottenere un voucher dalla PDND che successivamente devono inviare all'erogatore insieme alla normale richiesta di servizio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, registrandolo sulla PDND
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND».

Fruizioni > PetStoreAuthPDND@Ente v1 > Connettore

Connettore

Note: (*) Campi obbligatori

Connettore

Endpoint * ⓘ

Autenticazione Token

AutenticazioneHttps

Proxy

Ridefinisci Tempi Risposta

Autenticazione Token

Policy * ▾

Fig. 3.122: Associazione della Token Policy di Negoziazione al connettore

Fruizioni > PetStoreAuthPDND@Ente v1 > Configurazione > Proprietà

Proprietà

Visualizzati record [1-1] su 1

	Nome	Valore
<input type="checkbox"/>	<u>PDND-resource</u>	PetStore

Fig. 3.123: Proprietà “PDND-resource”

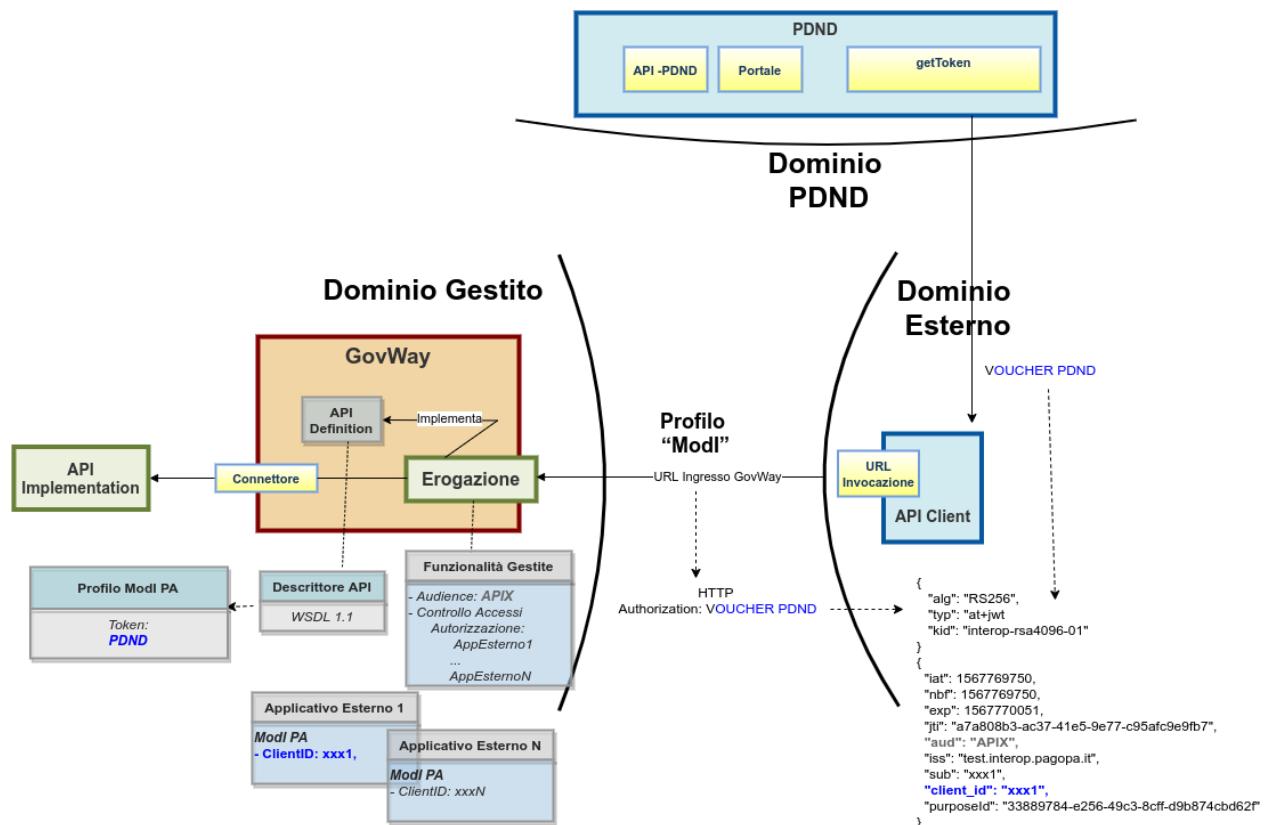


Fig. 3.124: Erogazione di una API SOAP con profilo “ModI”, pattern ID_AUTH_REST_01 via PDND

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.125: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API SOAP di esempio (Credit Card Verification) definita con pattern di interazione Bloccante e pattern di sicurezza «ID_AUTH_CHANNEL_01» e «ID_AUTH_REST_01 via PDND»;
- un'istanza Govway per la gestione del profilo ModI nel dominio dell'erogatore;
- un'authorization server che simula la PDND;
- un client del dominio esterno che invoca l'azione di esempio «CheckCC» dell'erogazione esposta da Govway;
- il server “Credit Card Verification” di esempio che riceve le richieste inoltrate dal Govway e produce le relative risposte. Per questo scenario viene utilizzato il server disponibile on line all'indirizzo “<https://ws.cdyne.com/creditcardverify/luhnchecker.asmx>”.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth+PDND - IN App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console govwayMonitor.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*.

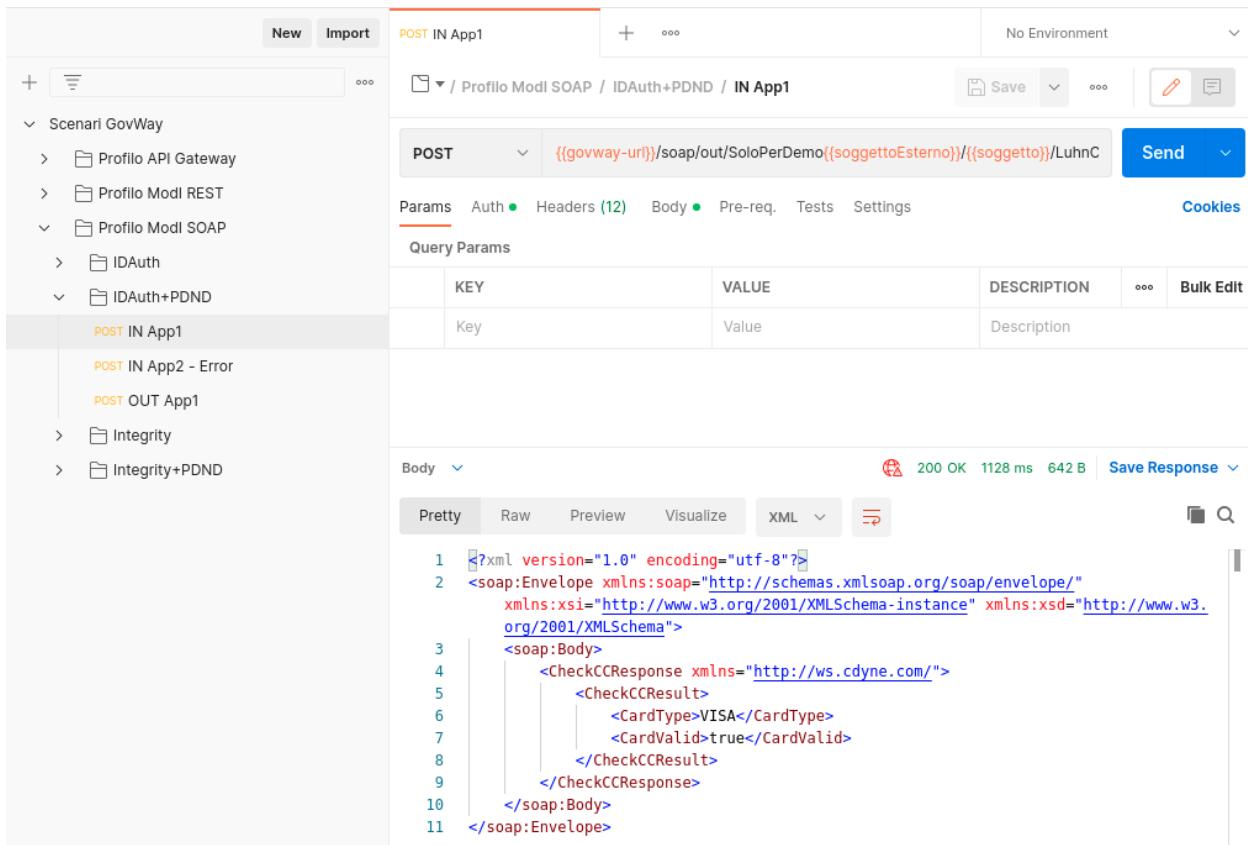
Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo “App1-ModI” identificato grazie al claim “client_id” presente all'interno del token. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth+PDND - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



The screenshot shows the Postman application interface. On the left, the sidebar lists scenarios: Scenari GovWay, Profilo API Gateway, Profilo Modl REST, Profilo Modl SOAP (expanded to show IDAuth and IDAuth+PDND), POST IN App1 (selected), POST IN App2 - Error, POST OUT App1, Integrity, and Integrity+PDND. The main area shows a POST request to `{{govway-url}}/soap/out/SoloPerDemo{{soggettoEsterno}}/{{soggetto}}/LuhnC`. The 'Params' tab is selected, showing a table with a single row: Key (Value) and Value (Value). The 'Body' tab shows a Pretty-printed XML response:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.
   org/2001/XMLSchema">
3   <soap:Body>
4     <CheckCCResponse xmlns="http://ws.cdyne.com/">
5       <CheckCCResult>
6         <CardType>VISA</CardType>
7         <CardValid>true</CardValid>
8       </CheckCCResult>
9     </CheckCCResponse>
10    </soap:Body>
11  </soap:Envelope>

```

The status bar at the bottom right indicates 200 OK, 1128 ms, 642 B, and a Save Response button.

Fig. 3.126: Pattern IDAuth+PDND - Erogazione API SOAP, esecuzione da Postman



Fig. 3.127: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Esecuzione*. Nel seguito viene riporta solamente la differenza relativa alla registrazione dell'API.

Registrazione API

Viene registrata l'API «CreditCardVerificationAuthPDND» con il relativo descrittore WSDL. Viene selezionato il solo pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) mentre non deve essere selezionato alcun pattern di sicurezza messaggio nella sezione «ModI» poichè la gestione del token avverrà tramite validazione di un token OAuth attivato sull'erogazione (Fig. 3.128).

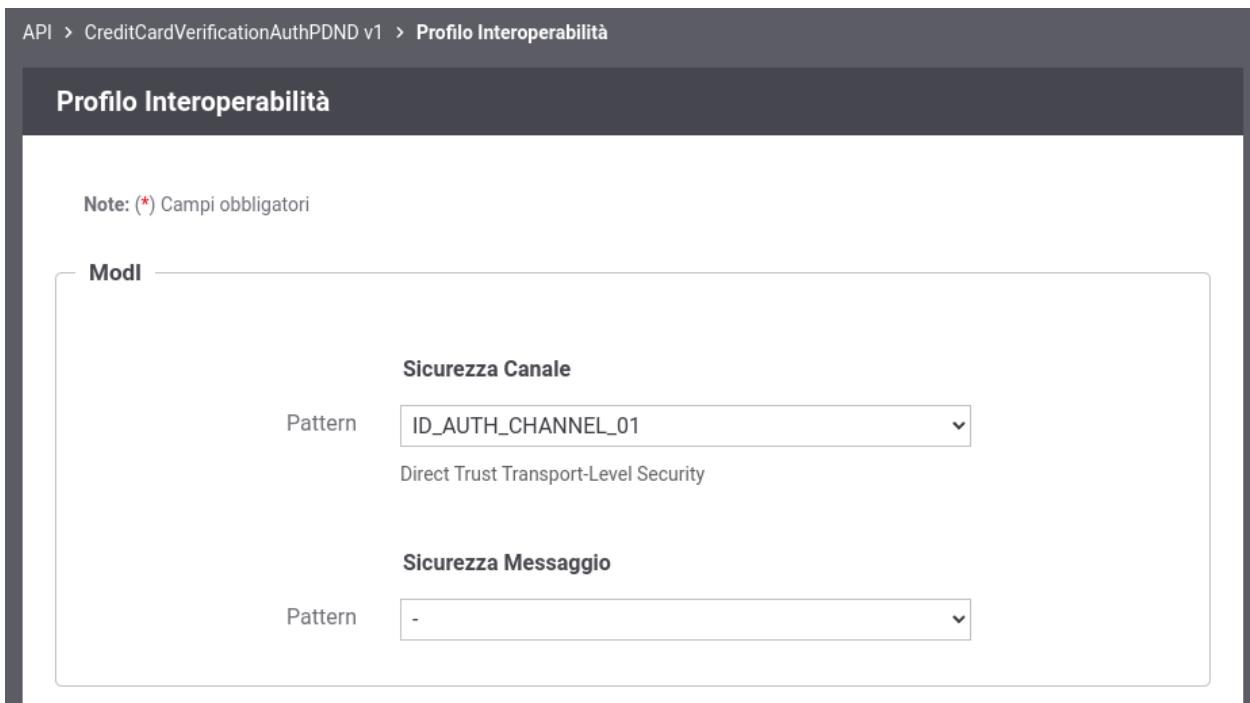


Fig. 3.128: Configurazione Pattern ModI con «ID_AUTH_CHANNEL_01» senza sicurezza messaggio

3.3.4 Fruizione API SOAP

Obiettivo

Fruire di un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd.

Sintesi

Mostriamo in questa sezione come procedere per l'integrazione di un applicativo con un servizio SOAP erogato nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui il servizio è stato registrato sulla PDND, e il fruitore per poterlo fruire

deve ottenere un voucher dalla PDND che successivamente deve inviare all'erogatore insieme alla normale richiesta di servizio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa registrata su PDND
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.130: Profilo ModI della govwayMonitor

L'esecuzione dello scenario si basa sui seguenti elementi:

- una API SOAP di esempio (Credit Card Verification) definita con pattern di interazione Bloccante e pattern di sicurezza «ID_AUTH_CHANNEL_01» e «ID_AUTH_REST_01 via PDND»;
- un'istanza Govway per la gestione del profilo ModI nel dominio del fruitore;
- un'authorization server che simula la PDND;
- un client del dominio gestito che invoca l'azione di esempio «CheckCC» sulla fruizione configurata su Govway.

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - IDAuth+PDND - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

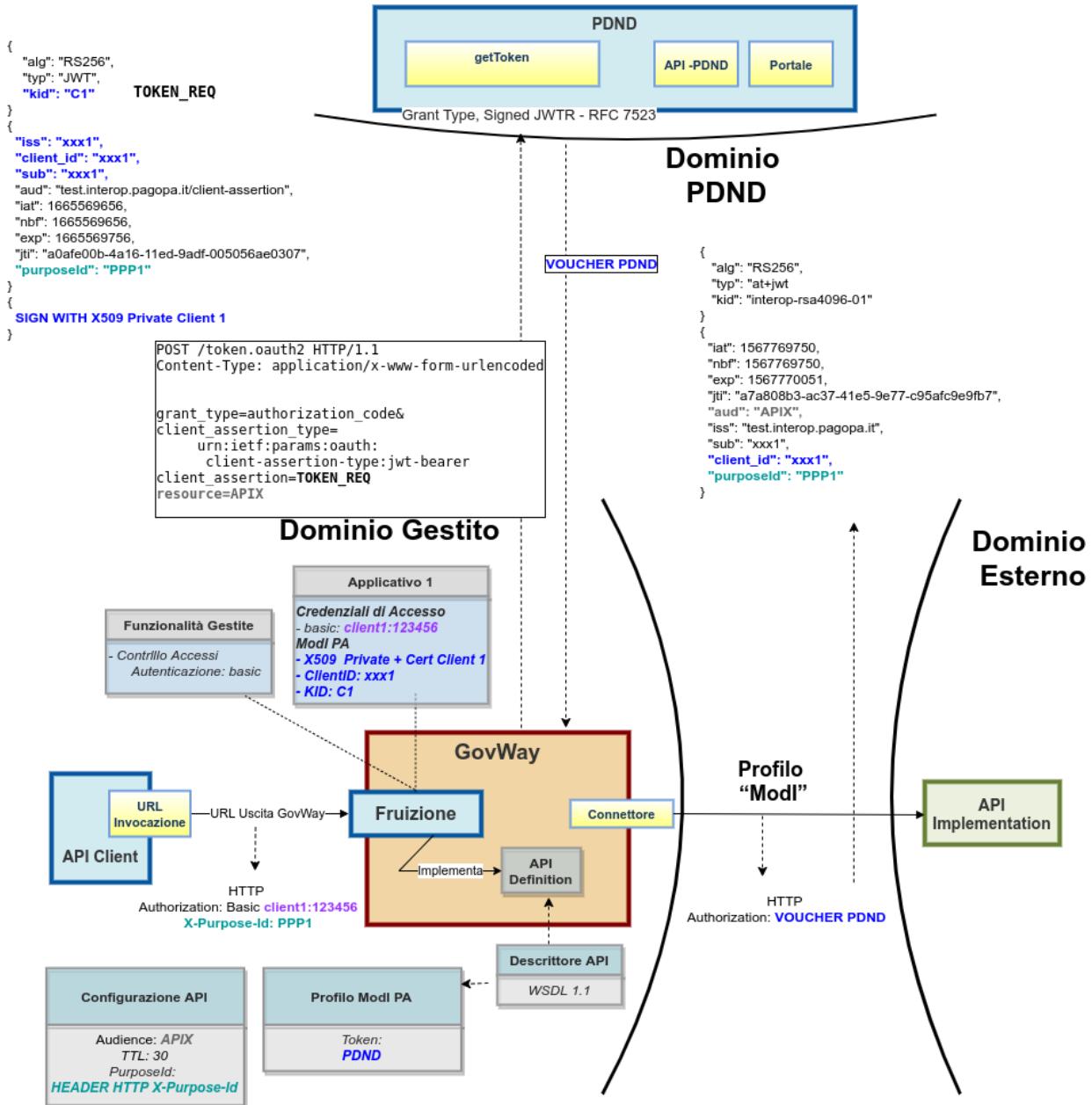


Fig. 3.129: Fruizione di una API SOAP con profilo "ModI", pattern ID_AUTH_REST_01 via PDND

Fig. 3.131: Pattern IDAuth+PDND - Fruizione API SOAP, esecuzione da Postman

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell’interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.132: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Configurazione*. Nel seguito viene riporta solamente la differenza relativa alla registrazione dell’API.

Registrazione API

Viene registrata l’API «CreditCardVerificationAuthPDND» con il relativo descrittore WSDL. Viene selezionato il solo pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) mentre non deve essere selezionato alcun pattern di sicurezza messaggio nella sezione «ModI» poichè la gestione del token avverrà tramite validazione di un token OAuth attivato sull’erogazione (Fig. 3.133).

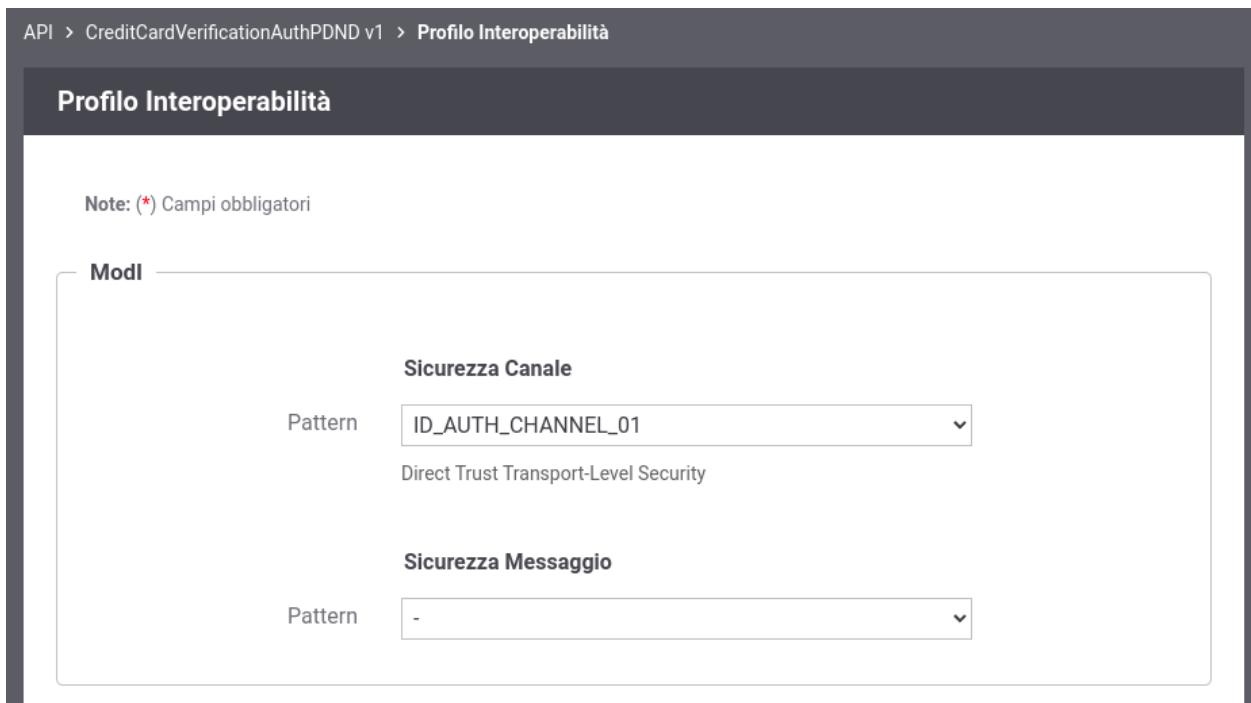


Fig. 3.133: Configurazione Pattern ModI con «ID_AUTH_CHANNEL_01» senza sicurezza messaggio

3.4 Pattern “ID_AUTH” via PDND + “INTEGRITY”

Gli scenari riportati in questa sezione riguardano API configurate con pattern modipa_pdnd_integrity.

3.4.1 Erogazione API REST

Obiettivo

Esporre un servizio, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd_integrity.

Sintesi

Mostriamo in questa sezione come procedere per l'esposizione di un servizio REST da erogare nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui un servizio è stato registrato sulla PDND, e i fruitori per poterlo fruire devono ottenere un voucher dalla PDND che successivamente devono inviare all'erogatore insieme alla normale richiesta di servizio. Oltre al voucher devono anche presentare il token di sicurezza «Agid-JWT-Signature» previsto dal pattern «INTEGRITY_REST_01» a garanzia dell'integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID. Il servizio viene registrato sulla PDND.
2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01» via PDND».
4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_REST_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.135: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario [Esecuzione](#) con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_REST_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity+PDND - IN App3» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

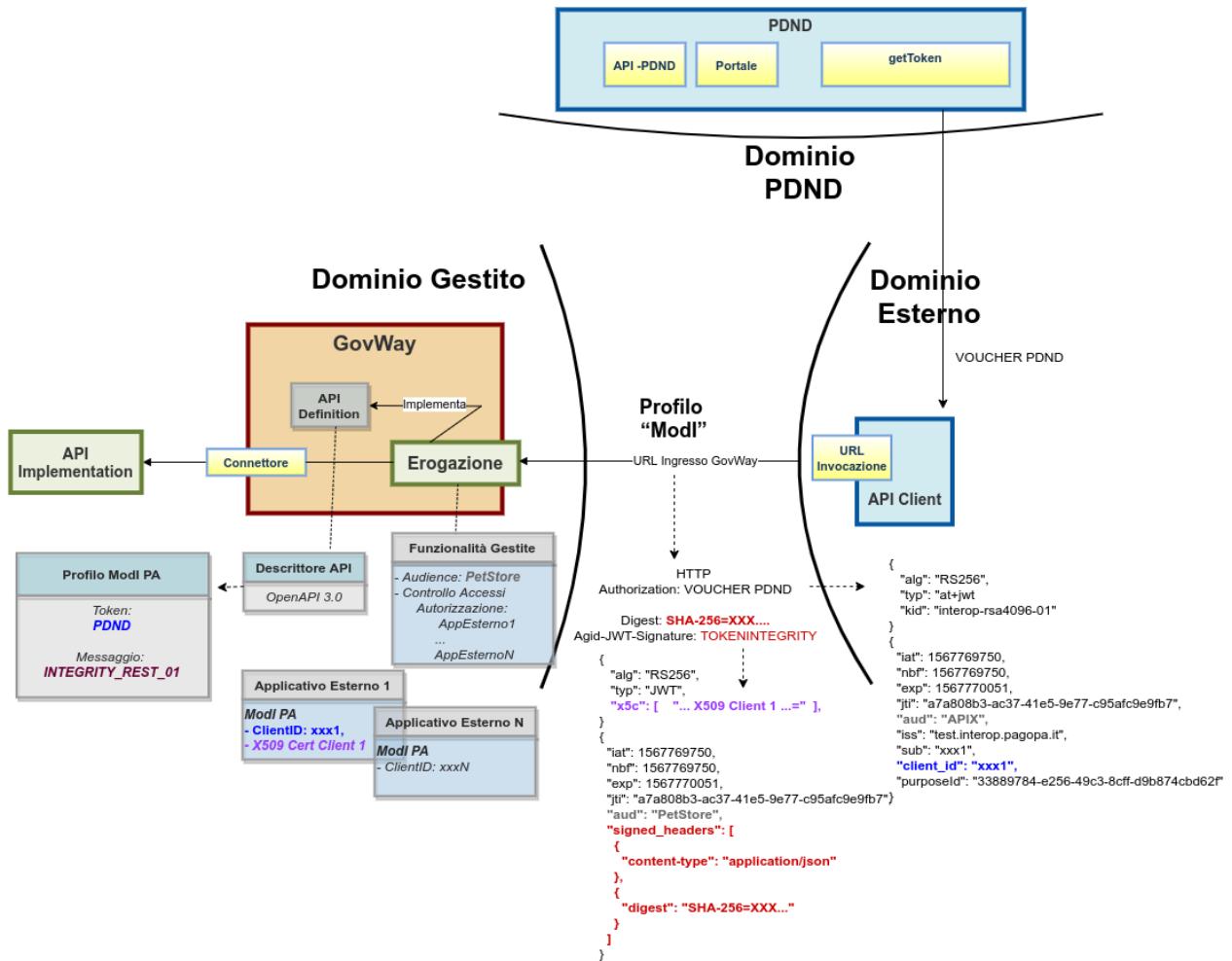


Fig. 3.134: Erogazione di una API REST con profilo “ModI”, pattern INTEGRITY_REST_01 e pattern ID_AUTH_REST_01 via PDND

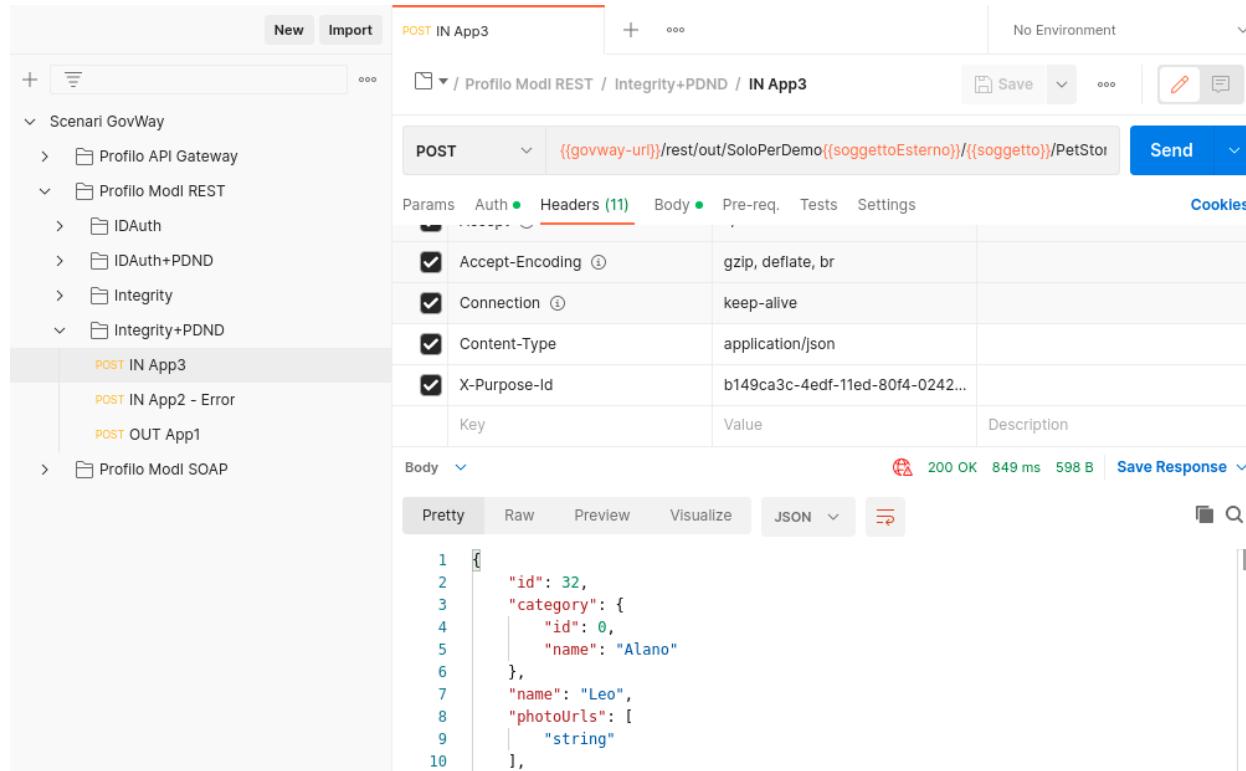


Fig. 3.136: Pattern Integrity+PDND - Erogazione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruttore, come in Fig. 3.137. Come si nota, al payload JSON è associato un insieme di header HTTP tra i quali «Authorization» e «Agid-Jwt-Signature» che contengono rispettivamente il token di sicurezza che il fruttore ha ottenuto dalla PDND e il token di integrità. È inoltre presente l'header http «Digest» che contiene il valore per la verifica dell'integrità del payload.
- Tutte le analisi che riguardano il token di autenticazione generato dalla PDND sono le medesime descritte nello scenario *Esecuzione*.
- Inoltre grazie alle configurazioni presenti nell'erogazione, ed in particolare alla relazione di trust stabilita con il fruttore, Govway è in grado di validare i dati di sicurezza ricevuti nel token «Agid-JWT-Signature». Nella fase di validazione del token si può notare come nella sezione header (Fig. 3.138) viene riportata l'identità del fruttore sotto forma di certificato X.509 a differenza di quello ottenuto dalla PDND.
- Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruttore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso e la corrispondenza del digest con il payload e con il valore interno al token «Agid-Jwt-Signature». Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore.

Le evidenze del processo di validazione relative al token PDND sono le medesime descritte nella scenario *Esecuzione*.

Headers	
Nome	
Content-Type	application/json
Govway-Message-Id	d1b37101-4fbb-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	0ab5fecb-2b64-497f-9a8e-ff0a6dbd24ab
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVClsImtpZC16ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6xWqdhfvHBaJT3on7jaCV6LVEXEaqAqfDWwI48L8SextE3UyuuGh-1s-g6320H8j6SIFr8tzsK4p-Fc94WcIxhMJxjXAer6Sh80
Agid-Jwt-Signature	eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVClsImtpZC16ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6WyJNSilVuNpGcBUWGoh1dKhKCv6nd6LFjWiFsdExxjto5i8lBtyjExSu06IHL0iaD2p1jkYrG37MgE6f-1xBYCqjElCchD6GQ8R4fEc5
Digest	SHA-256=OhjWochHmyIM/B4HeXiplNxygvqU7zKjERTUMDPVfhPY=
Accept	/*
Govway-Transaction-Id	d1a3b973-4fbb-11ed-a5ac-0242ac140002
Transfer-Encoding	chunked

Fig. 3.137: Messaggio inviato dal fruttore

```

HEADER: ALGORITHM & TOKEN TYPE

ID → {
  "alg": "RS256",
  "typ": "JWT",
  "kid": "app1.enteesterno.govway.org",
  "x5c": [
    "MIIE/jCCAuagAwIBAgICAPgwDQYJKoZIhvcNAQELBQAwNjELMAkGA1UEBhMCAxQxEzARBgNVBAoMCmdvdndheS5vcmcxExJABgNVBAMMCUdvd1dheSBDQTAeFw0yMjEwMTkwNzU1NThaFw0zNzEwMTUwNzU1NThaMEgxCzAJBgNVBAYTAm10MRMwEQYDVQQDApnb3Z3YXkub3JnMSQwIgYDVQQDDBthcHAxLmVudGVFc3R1cm5vLmdvdndheS5vcmcwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC1/cfENX06hdvEVxJiJAF00ePjn5Sh/HIJ2du8hRv0zA+KFFieaF4xh1mS0T1oq/vwdxFqvcd2k1bTJ37rjBo6DKuQZor83j/Do87x3sFJe/epGKx96Q3PRE9mA1qx3Y5FFShfGNzG2RFNA2jhVQ/bs8d9E051FC3XshF90CtJJs9LGvT2+0+uJK3siA6htKcYQ58UcK1W1Y109MnXqaz82TiH93eTSkk33w0A9atzC0w3JAVmcRRkd0hFBjMeEvNR86cdNfy9Xit7ZDR11IB8tel0/f1/oAW0oK/3TbF1X0rVL1QhMc1JdqS3NwJLAyoqmZT/Xh5DqjDi7ldghwbAgMBAAGjggECMIH/MAkGA1UdEwQCMAAwEQYJYIZIAYb4QgEBBAQDAgeAMDMGCWCGSAGG+EIBDQQmFiRPcGVuU1NMIEd1bmVyYXR1ZCBDbG11bnQgQ2VydG1maWNhdGUwHQYDVR0OBYEFCBwk8Bs9JS+6c/vTU+JX0eqX81BMGYGA1UdIwRfMF2AFCqHFNpm2RdIA3igRXzNEeJ5ivegoTqk0DA2MQswCQYDVQQGEwJpdDETMBEGA1UECgwKZ292d2F5Lm9yZzESMBAGA1UEAwJR292V2F5IENBggkA4tGAdmeSJF4wDgYDVR0PAQH/BAQDAgXgMBMGA1UdJQQMMAoGCsGAQUFBwMCMA0GCSqGSIb3DQEBCwUAA4ICAQDRj52cdYwcqFDNmC29CY0DR0N0TM/5RKq9sL6sgI7z4cUmkyIeGh/9YQDoRFhDBVGZ80rx0kasZ/Po0Iuw+41f9IDTBe04Ym0CK3M1M9H2LiEKe9hngRtjzGw5tFRQnqKbLLX61otJAXuE488SrSAMbEdez1bZt+V1Sgc48f0KsjShUs8CwSW0G6RE5w4Q4oa0dX971PTziWDoFnxBfN17/HAYA0625/vcp8PrZLqhTIGH7dt+1T4Hb+i10wKBS7B8Cab0Gh0spiHDDgNEYX50d1ZYmWJQ10ysK61Yx1WtCrKPfmsvSeqiVxJPHUgwTsFPrgoVRt+dT1NnAdXYxFk0Yxz7zn7qeKDi6cXHLTsYet1cQfedYDPE0rli4GFL1KY37NFqRtJx5NadkJk6GXk43zIFQo119PGJ8nVHupB6IBJ1h/6xem1TTMSt52zcjV4b5zRHL8ZqNF+S0QnJKcH2FcyAYuGjuVj0qa5rhi5wNcy7ilcDShM8tsPJ5qpW1ME0mhmWY+w5KBCpMoLBn9cvqAn/N19L3e3SqH1Klp8Hw05CtH4/tLEe3N+0z+8xzcmLdqbaZ9nD7YVLVoyt5Y+Ixuj17F18dzEh9dzLhJojsBmPjoFMMyulbpjZG0A1TjKVpkxyXgaqsd9Hjs4ATg79Vk8U/GnEXJhXQxU2TYw==",
  ],
  "x5t#S256": "agRQxqs-VYDP2NIzbR7XH2GiInWH2bcL1xMPhimfMKK"
}

```

Fig. 3.138: Sezione «Header» del Token di sicurezza «Agid-Jwt-Signature»

Nel payload del token «Agid-JWT-Signature» (Fig. 3.139) sono invece presenti i riferimenti temporali (iat, nbf, exp), l'audience (aud) e il claim «signed_headers» utilizzato per la verifica dell'integrità.

PAYLOAD: DATA
<pre>{ "iat": 1666190361, "nbf": 1666190361, "exp": 1666190421, "jti": "d1b37101-4fbb-11ed-a5ac-0242ac140002", "aud": "petstore.ente.govway.org", "client_id": "app1.enteesterno.govway.org", "iss": "SoloPerDemoEnteEsterno", "sub": "SoloPerDemoFirmatarioApp1", "signed_headers": [{ "digest": "SHA- 256=0hjWocHmy1M/B4HeXlplNxygvqU7zKjERTUMDPVfhPY=", "content-type": "application/json" }] }</pre>

Fig. 3.139: Sezione «Payload» del Token di sicurezza «Agid-Jwt-Signature»

Le evidenze del processo di validazione relativo al pattern «INTEGRITY_REST_01» sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.140). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dai token di sicurezza presenti, tra cui si può notare il digest e gli header http firmati.

Informazioni ModI

Sicurezza Messaggio INTEGRITY_REST_01 con ID_AUTH_REST_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Accesso CRUD

Sicurezza Messaggio

Digest SHA-256=OhjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=

ClientId app1.enteesterno.govway.org

Subject SoloPerDemoFirmatarioApp1

Issuer SoloPerDemoEnteEsterno

MessageId d1b37101-4fbb-11ed-a5ac-0242ac140002

Audience petstore.ente.govway.org

NotBefore 2022-10-19_16:39:21.000

Expiration 2022-10-19_16:40:21.000

IssuedAt 2022-10-19_16:39:21.000

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Headers HTTP Firmati

content-type application/json

digest SHA-256=OhjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=

Fig. 3.140: Traccia della richiesta elaborata dall’erogatore

- Lo scenario è preconfigurato per autorizzare puntualmente l’applicativo “App3-ModI” identificato grazie al claim “client_id” presente all’interno del token della PDND. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity+PDND - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. La sicurezza messaggio applicata è quella dei pattern «ID_AUTH_REST_01 via PDND» + «INTEGRITY_REST_01» come ampiamente mostrato precedentemente dove sono stati mostrati i token validati e i criteri autorizzativi.

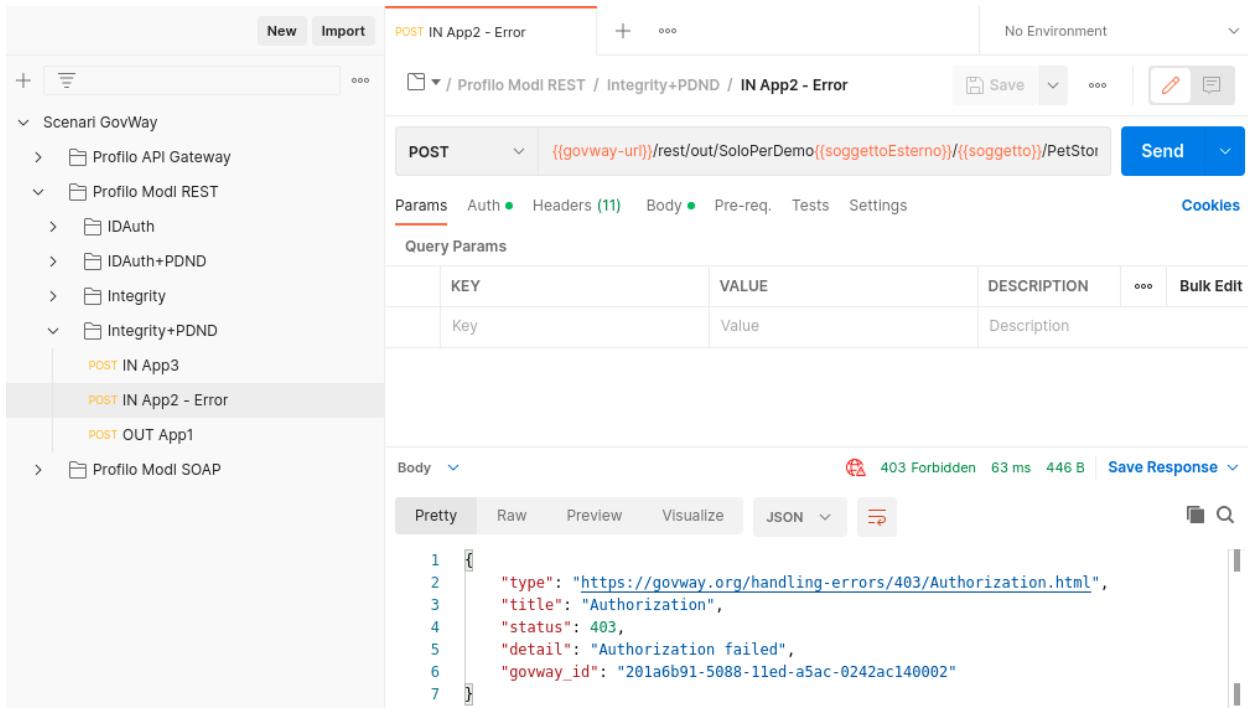


Fig. 3.141: Pattern Integrity+PDND - Erogazione API REST - Autorizzazione negata, esecuzione da Postman

2. L'identificazione del fruitore avviene rispetto al claim "client_id" presente all'interno del token. È stato anche mostrato come sia possibile configurare criteri autorizzativi puntuali.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità "ModI". Si suggerisce inoltre di selezionare il soggetto "Ente" per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.142: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario *Configurazione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_REST_01».

Registrazione API

Viene registrata l'API «PetStoreIntegrityPDND» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) e «INTEGRITY_REST_01» con ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.56). Viene inoltre indicato di utilizzare il solo header HTTP "Agid-JWT-Signature".

API > PetStoreIntegrityPDND v1 > Profilo Interoperabilità

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_01

Direct Trust Transport-Level Security

Sicurezza Messaggio

Pattern: INTEGRITY_REST_01 con ID_AUTH_REST_01

Integrità payload del messaggio

Header HTTP del Token: Agid-JWT-Signature

Applicabilità: Richiesta e Risposta

Digest Richiesta: Non ripudiabilità della trasmissione i

Informazioni Utente: Dati dell'utente che effettua la richiesta i

Fig. 3.143: Configurazione Pattern ModI «INTEGRITY_REST_01 con ID_AUTH_REST_01» sulla API REST con utilizzo del solo header HTTP “Agid-JWT-Signature”

Applicativo Esterno

È opzionalmente possibile registrare l'applicativo esterno che corrisponde al fruitore del servizio come descritto nello scenario nello scenario [Configurazione](#).

La registrazione comporta l'associazione all'applicativo sia del "client_id" necessario all'identificazione che del certificato di firma che verrà atteso nell'header HTTP "Agid-JWT-Signature" (Fig. 3.144). Questo scenario è quello preconfigurato.

Erogazione

Nell'erogazione «PetStoreIntegrityPDND», relativa all'API precedentemente inserita, vanno indicati i dati specifici nella sezione «ModI Richiesta» (Fig. 3.145) necessari per validare le richieste in ingresso relativamente al token "Agid-JWT-Signature".

La sezione «ModI Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza "Agid-JWT-Signature" da inserire nel messaggio di risposta (Fig. 3.146).

3.4.2 Fruizione API REST

Obiettivo

Fruire di un servizio REST, definito tramite una API REST (OpenAPI 3.0), accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd_integrity.

Sintesi

Mostriamo in questa sezione come procedere per l'integrazione di un applicativo con un servizio REST erogato nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui il servizio è stato registrato sulla PDND, e il fruitore per poterlo fruire deve ottenere un voucher dalla PDND che successivamente deve inviare all'erogatore insieme alla normale richiesta di servizio. Oltre al voucher il fruitore devo anche presentare il token di sicurezza «Agid-JWT-Signature» previsto dal pattern «INTEGRITY_REST_01» a garanzia dell'integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa e pubblicata su PDND.
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND».
4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_REST_01»

Applicativo

Dominio	Esterno
Soggetto	EnteEsterno
Nome *	<input type="text" value="App3-PDND"/>
Tipo	Client
Proprietà(0)	

Ruoli

visualizza(0)

Modi

Sicurezza Messaggio	<input type="text" value="Authorization PDND + Integrity"/> ▼
Certificato	
Cambia Certificato	
Aggiungi Certificato	
Download	
Verifica	<input checked="" type="checkbox"/>
Subject	<input type="text" value="/c=it/cn=app3.enteEsterno.govway.org/o=govway.org/"/>
Issuer	<input type="text" value="/c=it/cn=GovWay CA/o=govway.org/"/>
Serial Number	250 (Hex) 00:FA
Self Signed	No
Not Before	20/10/2022 09:45:00
Not After	16/10/2037 09:45:00
ClientId registrato sulla PDND	
Token Policy *	<input type="text" value="PDND"/> ▼
Identificativo *	<input type="text" value="App3-Esterno-PDND"/>

Fig. 3.144: Configurazione applicativo esterno (fruitore)

Modi - Richiesta

Sicurezza Messaggio

Riferimento X.509
x5c (Certificate)
x5t#256 (Certificate SHA-256 Thumbprint)
x5u (URL)

TrustStore Certificati
Default

Time to Live
Default

Audience
petstore.ente.govway.org

Se non viene fornito un valore, il valore atteso all'interno del security token corrisponderà all'url di invocazione

Fig. 3.145: Configurazione richiesta dell'erogazione

Modi - Risposta

Sicurezza Messaggio

Algoritmo
RS256

HTTP Headers da firmare *
Digest x Content-Type x Content-Encoding x

Riferimento X.509
Utilizza impostazioni della Richiesta

Certificate Chain

KeyStore
Default

Time to Live (secondi) *
60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Claims
 ⓘ

Indicare per riga i claims (nome=valore); visualizzare 'info' per maggiori dettagli

Fig. 3.146: Configurazione risposta dell'erogazione

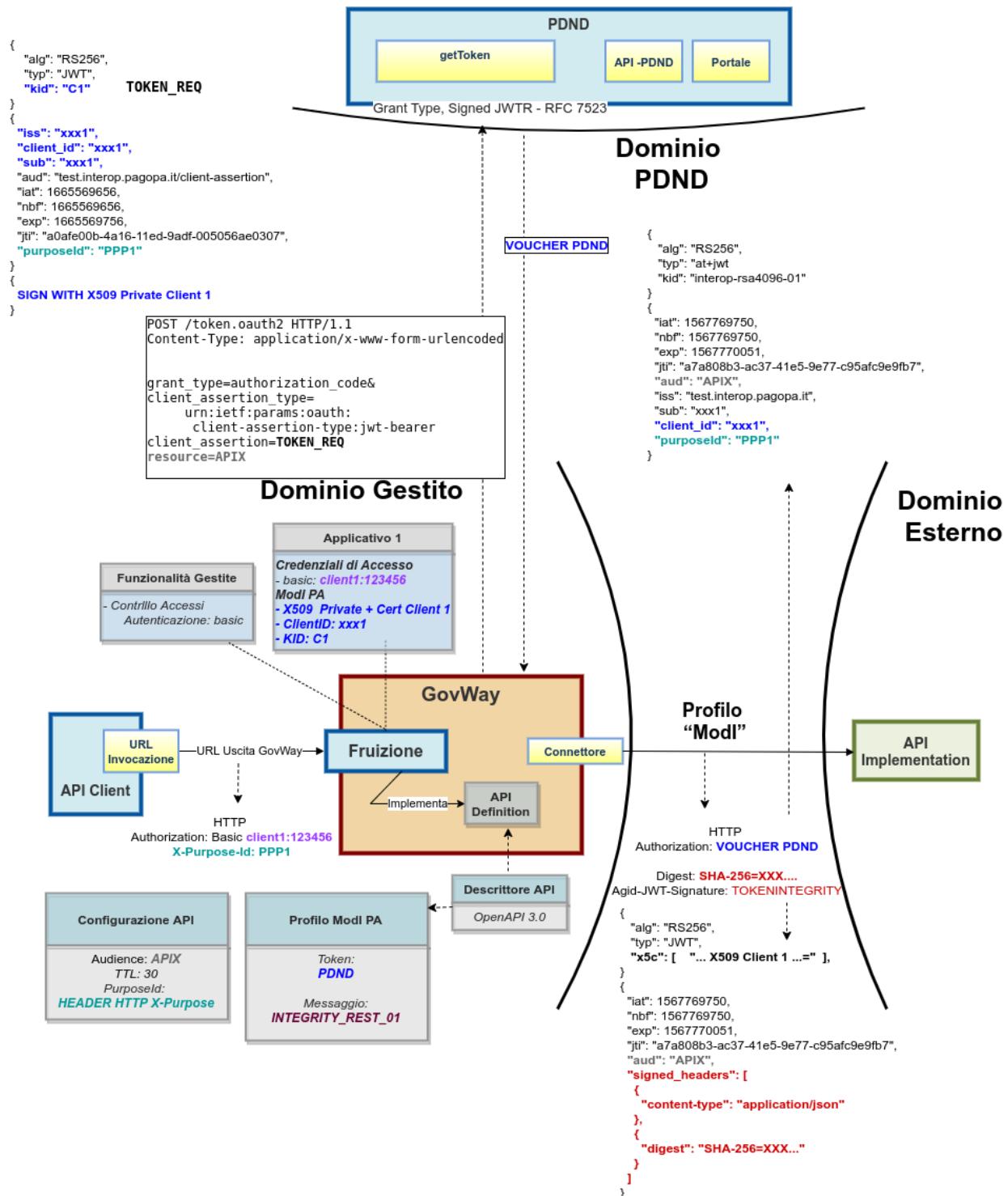


Fig. 3.147: Fruizione di una API REST con profilo "ModI", pattern INTEGRITY_REST_01 e pattern ID_AUTH_REST_01 via PDND

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.148: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_REST_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI REST - Integrity+PDND - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

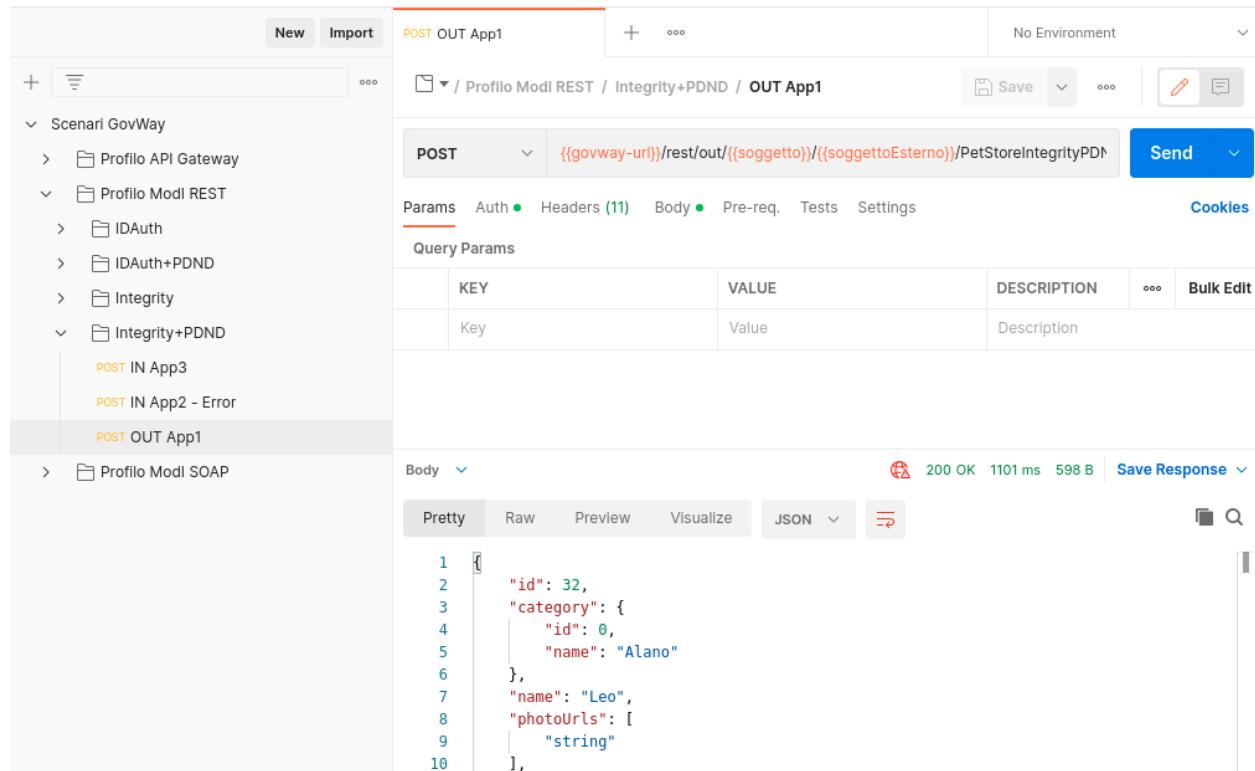


Fig. 3.149: Pattern Integrity+PDND - Fruizione API REST, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto nelle diverse fasi dell'esecuzione andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Il messaggio di richiesta inviato dal fruitore viene elaborato da Govway che, tramite la configurazione della firma digitale associata all'applicativo mittente, è in grado di produrre un token di sicurezza da inviare alla PDND con

il quale ottenere indietro un voucher spendibile per il servizio desiderato. Questa parte è stata ampiamente mostrata nella scenario *Esecuzione*.

Oltre al token della PDND, GovWay produce un ulteriore token di sicurezza «Agid-JWT-Signature» previsto dal pattern «INTEGRITY_REST_01». Da govwayMonitor si può visualizzare il messaggio di richiesta in uscita che è il medesimo di quello in entrata con la differenza che sono stati aggiunti gli header HTTP «Authorization» e «Agid-Jwt-Signature» che contengono rispettivamente il token ottenuto dalla PDND e il token dell'integrità. È inoltre presente l'header http «Digest» che contiene il valore utilizzabile dall'erogatore per la verifica dell'integrità del payload. (Fig. 3.150).

Headers	
Nome	
Content-Type	application/json
Govway-Message-Id	d1b37101-4fbb-11ed-a5ac-0242ac140002
X-Forwarded-Server	411885f186f6
X-Real-Ip	172.20.0.1
Postman-Token	0ab5fecb-2b64-497f-9a8e-ff0a6dbd24ab
X-Forwarded-For	172.20.0.2
Cache-Control	no-cache
Authorization	Bearer eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6xWqdhfvHBaJT3on7jaCV6LVEXEaqAqfDWwI48L8SextE3UyuuGh-1s-g6320H8j6SIf8tzsK4p-Fc94WclxhMJxjXAer6Sh8C
Agid-Jwt-Signature	eyJhbGciOiJSUzI1NilsInR5cCl6IkpxVCIsImtpZCI6ImFwcDEuZW50ZWVzdGVybm8uZ292d2F5Lm9yZylsIng1Yyl6WyJNSjliVUUnPcBUWGoh1dKhKCv6nd6LFjWiFSdExxjto5i8iBtyjExSu06IHL0iaD2pI1jkYrG37MgE6f-1xBYCqjElCchD6GQ8R4fEc5
Digest	SHA-256=OhjWocHmylM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=
Accept	*/*
Govway-Transaction-Id	d1a3b973-4fbb-11ed-a5ac-0242ac140002
Transfer-Encoding	chunked

Fig. 3.150: Messaggio di richiesta in uscita (con token di sicurezza inseriti nell'header HTTP)

- L'header e i payload del token «Agid-JWT-Signature» sono identici a quelli già visualizzati nello scenario di erogazione REST, relativamente al messaggio in ingresso (Fig. 3.138 e Fig. 3.139). Le informazioni inserite nel token vengono anche tracciate e sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.151). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dai token di sicurezza, tra cui si può notare il digest e gli header http firmati.

Conformità ai requisiti ModI

Informazioni Modelli	
Sicurezza Messaggio	INTEGRITY_REST_01 con ID_AUTH_REST_01
Sicurezza Canale	ID_AUTH_CHANNEL_01
Interazione	Accesso CRUD
Sicurezza Messaggio	
X509-Issuer	CN=GovWay CA, O=govway.org, C=it
X509-Subject	CN=app1.ente.govway.org, O=govway.org, C=it
Digest	SHA-256=OhjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=
Subject	App1-PDND
Issuer	Ente
ClientId	Ente/App1-PDND
Audience	petstore.enteEsterno.govway.org
MessageId	d59e4915-508b-11ed-a5ac-0242ac140002
Expiration	2022-10-20_17:29:23.000
NotBefore	2022-10-20_17:28:23.000
IssuedAt	2022-10-20_17:28:23.000
Headers HTTP Firmati	
content-type	application/json
digest	SHA-256=OhjWocHmyIM/B4HeXlpINxygvqU7zKjERTUMDPVfhPY=

Fig. 3.151: Traccia della richiesta generata dal fruttore

I requisiti iniziali, legati alla comunicazione basata su uno scenario ModI, sono verificati dalle seguenti evidenze:

1. Viene effettuata una negoziazione del voucher PDND come mostrato nelle tracce relative ai token scambiati con la PDND.
2. L'invocazione del servizio avviene fornendo il voucher della PDND precedentemente negoziato.
3. Vengono inoltre prodotti gli header http «Agid-Jwt-Signature» e «Digest» previsti dal pattern di sicurezza «INTEGRITY_REST_01»

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.152: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario [Configurazione](#) con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_REST_01».

Registrazione API

Viene registrata l'API «PetStoreIntegrityPDND» con il relativo descrittore OpenAPI 3. Vengono selezionati i pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) e «INTEGRITY_REST_01» con ID_AUTH_REST_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.153). Viene inoltre indicato di utilizzare il solo header HTTP “Agid-JWT-Signature”.

Fruizione

Nella fruizione «PetStoreIntegrityPDND», relativa all'API precedentemente inserita, vanno indicati i dati specifici nella sezione «ModI Richiesta» (Fig. 3.154) necessari a generare il token “Agid-JWT-Signature”. In particolare è possibile specificare l'audience atteso dall'erogatore e il tempo di validità del token.

La sezione «ModI Risposta» definisce i criteri per la validazione del token di sicurezza “Agid-JWT-Signature” presente nel messaggio di risposta, come il truststore per l'autenticazione dell'erogatore (Fig. 3.155).

3.4.3 Erogazione API SOAP

Obiettivo

Esporre un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd_integrity.

Sintesi

Mostriamo in questa sezione come procedere per l'esposizione di un servizio SOAP da erogare nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui un servizio è stato registrato sulla PDND, e i fruitori per poterlo fruire devono ottenere un voucher dalla PDND che successivamente devono inviare all'erogatore insieme alla normale richiesta di

API > PetStoreIntegrityPDND v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_01

Direct Trust Transport-Level Security

Sicurezza Messaggio

Pattern: INTEGRITY_REST_01 con ID_AUTH_REST_01

Integrità payload del messaggio

Header HTTP del Token: Agid-JWT-Signature

Applicabilità: Richiesta e Risposta

Digest Richiesta: Non ripudiabilità della trasmissione i

Informazioni Utente: Dati dell'utente che effettua la richiesta i

Fig. 3.153: Configurazione Pattern ModI «INTEGRITY_REST_01 con ID_AUTH_REST_01» sulla API REST con utilizzo del solo header HTTP “Agid-JWT-Signature”

Modi - Richiesta

Sicurezza Messaggio

Algoritmo	RS256
HTTP Headers da firmare *	Digest <input checked="" type="checkbox"/> Content-Type <input checked="" type="checkbox"/> Content-Encoding <input checked="" type="checkbox"/>
Riferimento X.509	x5c (Certificate) x5t#256 (Certificate SHA-256 Thumbprint) x5u (URL)
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	60
Indica la validità temporale, in secondi, a partire dalla data di creazione del security token	
Audience	petstore.enteEsterno.govway.org i
Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore	
Claims	i
Indicare per riga i claims (nome=valore); visualizzare 'info' per maggiori dettagli	

Fig. 3.154: Configurazione richiesta della fruizione

Modi - Risposta

Sicurezza Messaggio

Riferimento X.509	Utilizza impostazioni della Richiesta
TrustStore Certificati	Default
Time to Live	Default
Verifica Audience	<input checked="" type="checkbox"/> La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente
i	

Fig. 3.155: Configurazione risposta della fruizione

servizio. Oltre al voucher devono anche presentare il token di sicurezza «Agid-JWT-Signature» previsto dal pattern «INTEGRITY_SOAP_01» a garanzia dell'integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

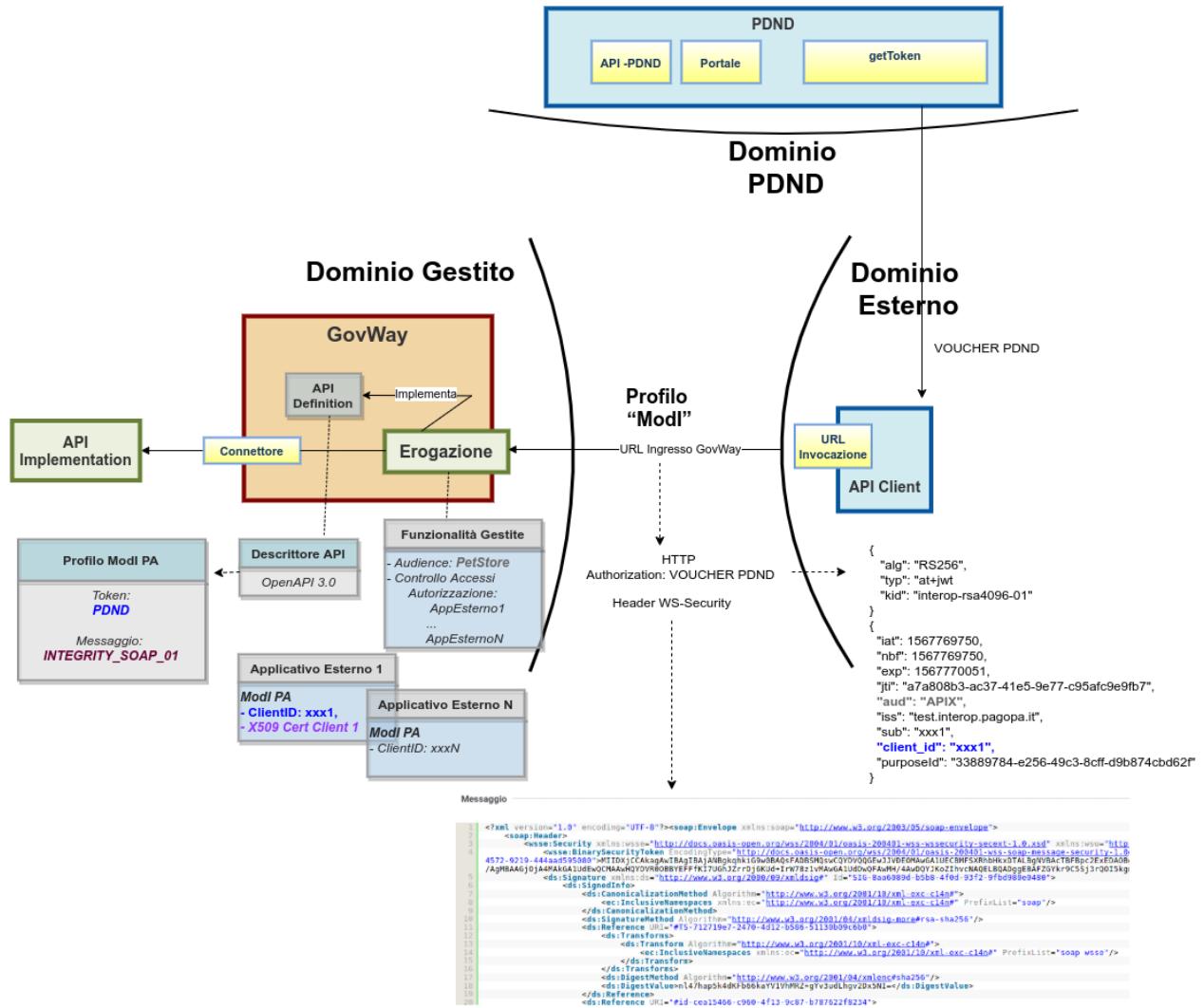


Fig. 3.156: Erogazione di una API SOAP con profilo “ModI”, pattern INTEGRITY_SOAP_01 e pattern ID_AUTH_REST_01 via PDND

Le caratteristiche principali di questo scenario sono:

1. Un applicativo eroga un servizio, rivolto a fruitori di domini esterni, in conformità al Modello di Interoperabilità AGID. Il servizio viene registrato sulla PDND.
 2. La comunicazione con i domini esterni avviene su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
 3. L'autenticità della comunicazione tra il servizio erogato e ciascun fruitore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND».
 4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_SOAP_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.157: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_SOAP_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity+PDND - IN App3» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

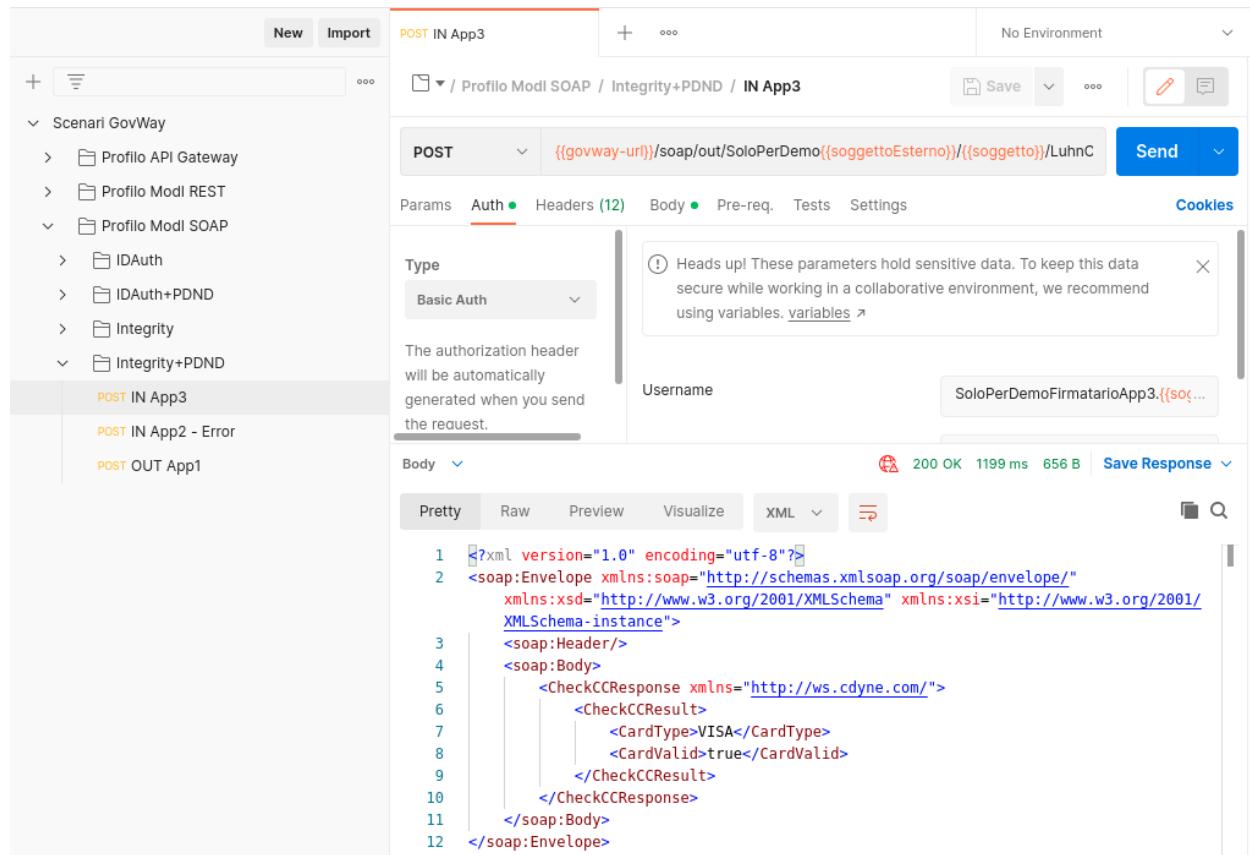


Fig. 3.158: Pattern Integrity+PDND - Erogazione API SOAP, esecuzione da Postman

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console govwayMonitor.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

- Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato dal fruitore, come in Fig. 3.32. Come si nota, il messaggio SOAP contiene nell'header WS-Security sia il token di sicurezza (elemento «BinarySecurityToken») sia l'elemento «WSAddressing - To» e il digest del payload (elemento «DigestValue») prodotti dal fruitore con la relativa firma digitale (elemento «SignatureValue») come previsto dal pattern «INTEGRITY_SOAP_01».

Messaggio

```

1  <?xml version="1.0" encoding="UTF-8"?><soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
2    <soap:Header>
3      <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://
4        <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-ws-soap-message-security-1.0#"
5          c7761d94d4f">MIIEzCIAueqAwIBAgICAN4wDQYJKoZIhvCNQaELBQAwNjELMAkGA1UEBhMCaX0xEzARBgNVBAoMdvdndheS5vcmcxE1AQBgNVBAMMCUdvlhdeSBQTAef
6          /Wu06/YXIV1DHLYmjypb/fI0SL8SKA6uW9swPxcogJPK9aqwgBv0/8w2Lpv1657H+BtNjLe8FhSmUnL7C25Hba/WivKh78213F5LYc4sY8H9nfC/fabQ0u0u1DltxWohKwzN
7          /ZAJBgNVHRMEAjAAMBEggWCGSAgg+EIBAQoEAWIHqDazBqLghkgBvhvCA00EJhYKT3BhbLNTTCBHZW51cmF0Zw0g02xpzW50IENlcnRpZmljYXRlMB0GA1Udbg0WBKRUAicYEN
8          /JIBWmVuatppwNcJRTz106qmIElqmoBTNLZ10MxI/+2sWQUTWNGNsU02z1TDs11rmeE1d1RcbKVvNcxtPHH4ysh5JdIp1fn7G3l4CtJHBHo2Ufu0eb03dFqqRc6QzMeEr
9          /OFgpiDpcA7fxITX0gDokm+WaqMAZ7s6DEmgW+h7KLkub0hvezwukbasdpYbqyciovDaomd4yWva15csvmubwSRIA1RH80uew0JcyeJ5fEY8fslFud0BLG934tI4Hnt2CBM8
10         /NKL76fLQPRGActEV4x0nvCe8NWm28oAPIohYpPUTv5Y1P5Y=</wsse:BinarySecurityToken>
11         <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-4bbe4224-d2df-4f57-814c-2b8a47ec328d">
12           <ds:SignedInfo>
13             <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
14               <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap"/>
15             </ds:CanonicalizationMethod>
16             <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
17             <ds:Reference URI="#TS-91e2766f-c512-4440-bfa1-046bbdec9b7">
18               <ds:Transforms>
19                 <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
20                   <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soap wsse"/>
21                 </ds:Transform>
22               </ds:Transforms>
23             </ds:Reference>
24           </ds:SignedInfo>
25           <ds:SignatureValue>...</ds:SignatureValue>
26         </ds:Signature>
27       <wsa:To>...</wsa:To>
28     </soap:Header>
29     <soap:Body>...</soap:Body>
30   </soap:Envelope>

```

Fig. 3.159: Messaggio inviato dal fruitore

- Tutte le analisi che riguardano il token di autenticazione generato dalla PDND sono le medesime descritte nello scenario *Esecuzione*.
- Il messaggio ricevuto dal Govway viene quindi validato, sulla base dei pattern di sicurezza previsti nello scambio, verificando in questo caso l'identità del fruitore, la validità temporale, la corrispondenza dell'audience ricevuto con quello atteso e la corrispondenza del digest rispetto al payload. Solo in caso di superamento dell'intero processo di validazione, il messaggio viene inoltrato al servizio erogatore. Le evidenze del processo di validazione sono visibili sulla govwayMonitor, andando a consultare la traccia del messaggio di richiesta (Fig. 3.160). Nella sezione «Sicurezza Messaggio» sono riportate le informazioni estratte dal token di sicurezza presente nell'header soap.
- Lo scenario è preconfigurato per autorizzare puntualmente l'applicativo “App3-ModI” identificato grazie al claim “client_id” presente all'interno del token della PDND. È possibile utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity+PDND - IN App2 - Error» per verificare che una richiesta proveniente da un differente applicativo non viene autorizzata.

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.

Informazioni Modl

Sicurezza Messaggio INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01

Sicurezza Canale ID_AUTH_CHANNEL_02

Interazione Bloccante

Sicurezza Messaggio

MessageId 13526172-4fc9-11ed-a5ac-0242ac140002

WSA-From app1.enteesterno.govway.org

WSA-To luhnCheckerSoap.ente.govway.org

Digest SHA256=sRq5LjK63zpG/FhfMWb/IE1HtNE2w1XYhHdLIWgxuX0=

Expiration 2022-10-19_18:15:14.957

IssuedAt 2022-10-19_18:14:14.957

X509-Issuer CN=GovWay CA, O=govway.org, C=it

X509-Subject CN=app1.enteEsterno.govway.org, O=govway.org, C=it

Elementi SOAP Firmati

Body http://schemas.xmlsoap.org/soap/envelope/

ReplyTo http://www.w3.org/2005/08/addressing

MessageID http://www.w3.org/2005/08/addressing

Action http://www.w3.org/2005/08/addressing

From http://www.w3.org/2005/08/addressing

To http://www.w3.org/2005/08/addressing

Timestamp http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd

Fig. 3.160: Traccia della richiesta elaborata dall'erogatore

The screenshot shows the Postman application interface. The left sidebar contains a tree view of scenarios and profiles, with 'Scenari GovWay' expanded to show 'Profilo API Gateway', 'Profilo Modl REST', 'Profilo Modl SOAP' (which is expanded to show 'IDAuth', 'IDAuth+PDND', 'Integrity', and 'Integrity+PDND'), and 'POST IN App2 - Error' selected. The main workspace shows a 'POST' request to the URL `{{govway-url}}/soap/out/SoloPerDemo{{soggettoEsterno}}/{{soggetto}}/LuhnC`. The 'Auth' tab is selected, showing 'Basic Auth' as the type. A note in the sidebar says: 'Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [variables](#)'.

The 'Body' tab shows the XML response, which is a SOAP fault message:

```

1  <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
2    <SOAP-ENV:Header/>
3    <SOAP-ENV:Body>
4      <SOAP-ENV:Fault>
5        <faultcode>SOAP-ENV:Client.Authorization</faultcode>
6        <faultstring xml:lang="en-US">Authorization failed</faultstring>
7        <faultactor>http://govway.org/integration</faultactor>
8        <detail>
9          <problem xmlns="urn:ietf:rfc:7807">
10            <type>https://govway.org/handling-errors/403/Authorization.html</type>
11            <title>Authorization</title>
12            <status>403</status>
13            <detail>Authorization failed</detail>
14            <govway_id>1d584f01-5091-11ed-a5ac-0242ac140002</govway_id>
15          </problem>
16        </detail>
17      </SOAP-ENV:Fault>

```

Fig. 3.161: Pattern Integrity+PDND - Erogazione API SOAP - Autorizzazione negata, esecuzione da Postman



Fig. 3.162: Profilo ModI della govwayConsole

Il processo di configurazione per questo scenario è del tutto analogo a quello descritto per lo scenario *Configurazione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_SOAP_01».

Registrazione API

Viene registrata l'API «CreditCardVerificationIntegrityPDND» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) e «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» (sicurezza messaggio) nella sezione «ModI» (Fig. 3.163).

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern: ID_AUTH_CHANNEL_01
Direct Trust Transport-Level Security

Sicurezza Messaggio

Pattern: INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01
Integrità payload del messaggio

Applicabilità: Richiesta e Risposta

Digest Richiesta: Non ripudiabilità della trasmissione (i)

Informazioni Utente: Dati dell'utente che effettua la richiesta (i)

Fig. 3.163: Configurazione Pattern ModI «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» sulla API SOAP

Erogazione

Nell'erogazione SOAP “LuhnCheckerSoapIntegrityPDND”, relativa all'API precedentemente inserita, vanno indicati i dati specifici nella sezione «ModI Richiesta» (Fig. 3.76) necessari per validare l'header WSSecurity previsto dal pattern «INTEGRITY_SOAP_01».

Modi - Richiesta

Sicurezza Messaggio

TrustStore Certificati	Default
Time to Live	Default
WSAddressing To	luhnCheckerSoap.ente.govway.org

Se non viene fornito un valore, il valore atteso all'interno del security token corrisponderà all'url di invocazione

Fig. 3.164: Configurazione richiesta dell'erogazione

La sezione «Modi Risposta» si utilizza per indicare i parametri per la produzione del token di sicurezza da inserire nel messaggio di risposta (Fig. 3.165).

Modi - Risposta

Sicurezza Messaggio

Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
KeyStore	Default
Time to Live (secondi) *	60

Indica la validità temporale, in secondi, a partire dalla data di creazione del security token della risposta

Fig. 3.165: Configurazione risposta dell'erogazione

3.4.4 Fruizione API SOAP

Obiettivo

Fruire di un servizio SOAP, definito tramite una interfaccia WSDL, accessibile in accordo al pattern di sicurezza descritto nella sezione modipa_pdnd_integrity.

Sintesi

Mostriamo in questa sezione come procedere per l'integrazione di un applicativo con un servizio SOAP erogato nel rispetto della normativa italiana alla base dell'interoperabilità tra i sistemi della pubblica amministrazione. In particolare andiamo ad illustrare lo scenario in cui il servizio è stato registrato sulla PDND, e il fruitore per poterlo fruire deve ottenere un voucher dalla PDND che successivamente deve inviare all'erogatore insieme alla normale richiesta di servizio. Oltre al voucher il fruitore devo anche presentare il token di sicurezza WSSecurity previsto dal pattern «INTEGRITY_SOAP_01» a garanzia dell'integrità del messaggio.

La figura seguente descrive graficamente questo scenario.

Le caratteristiche principali di questo scenario sono:

1. Un applicativo fruitore che dialoga con il servizio erogato in modalità ModI in accordo ad una API condivisa e pubblicata su PDND.
2. La comunicazione diretta verso il dominio erogatore veicolata su un canale gestito con il pattern di sicurezza canale «ID_AUTH_CHANNEL_01»
3. L'autenticità della comunicazione tra fruitore ed erogatore è garantita tramite sicurezza a livello messaggio con pattern «ID_AUTH_REST_01 via PDND».
4. L'integrità del messaggio scambiato è garantita tramite sicurezza messaggio aggiuntiva prevista nel pattern «INTEGRITY_SOAP_01»

Esecuzione

Nota: Al fine di avere una consultazione immediata delle informazioni di interesse per lo scenario si consiglia di impostare, nella console “govwayMonitor”, nel menù in alto a destra il Profilo di Interoperabilità “ModI”. Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le transazioni di interesse allo scenario e ignorare le transazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.167: Profilo ModI della govwayMonitor

L'esecuzione dello scenario è del tutto analogo a quello descritto nello scenario *Esecuzione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_SOAP_01».

Per eseguire e verificare lo scenario si può utilizzare il progetto Postman a corredo con la request «Profilo ModI SOAP - Integrity+PDND - OUT App1» che è stata preconfigurata per il funzionamento con le caratteristiche descritte sopra.

Dopo aver eseguito la «Send» e verificato il corretto esito dell'operazione è possibile andare a verificare cosa è accaduto, nel corso dell'elaborazione della richiesta, andando a consultare la console “govwayMonitor”.

Le verifiche da effettuare sono le medesime di quelle descritte nello scenario *Esecuzione*. Di seguito vengono riportati solo i punti salienti in cui emerge una differenza dovuta al pattern di sicurezza diverso utilizzato.

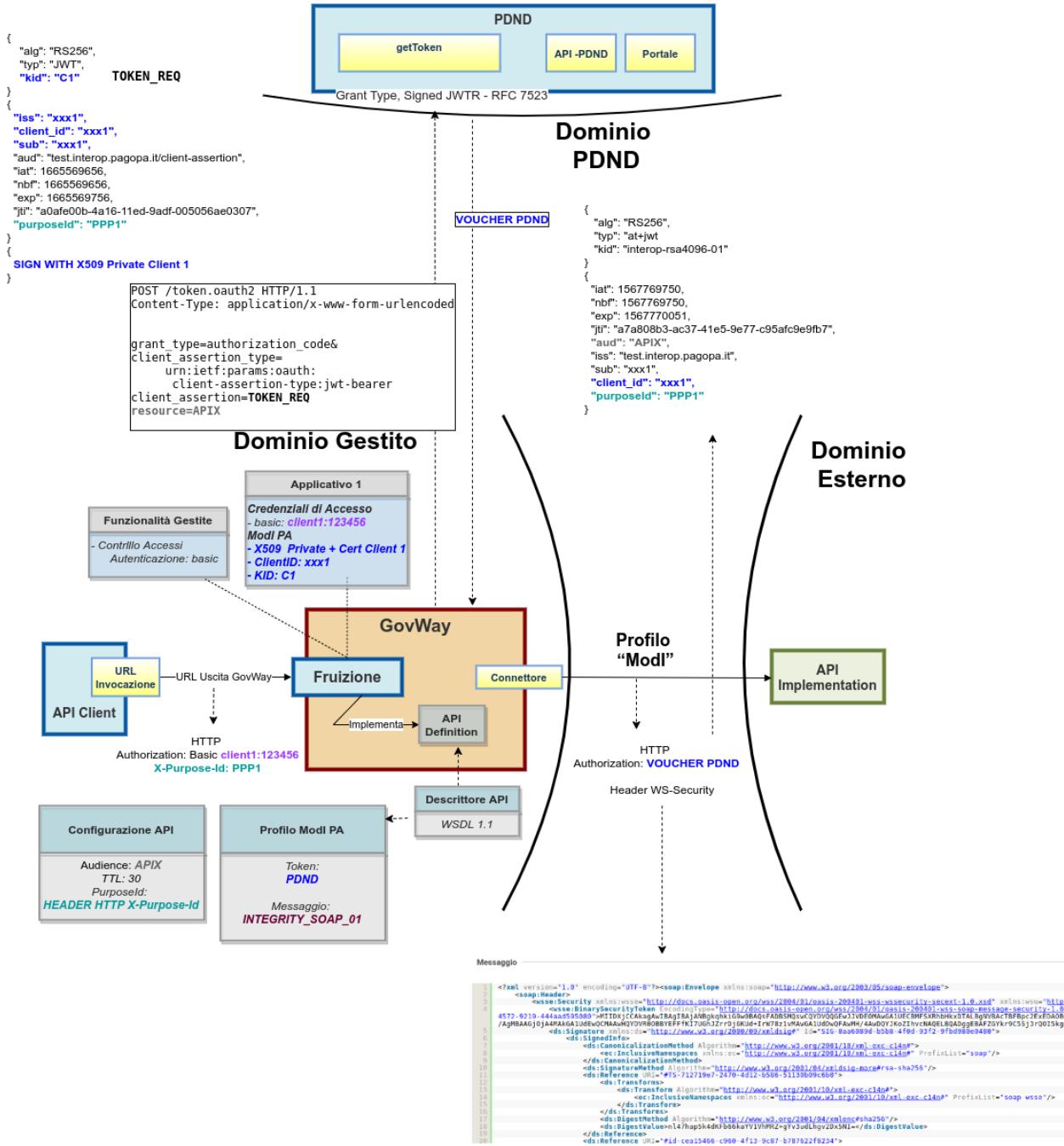


Fig. 3.166: Fruizione di una API SOAP con profilo "ModI", pattern INTEGRITY_SOAP_01 e pattern ID_AUTH_REST_01 via PDND

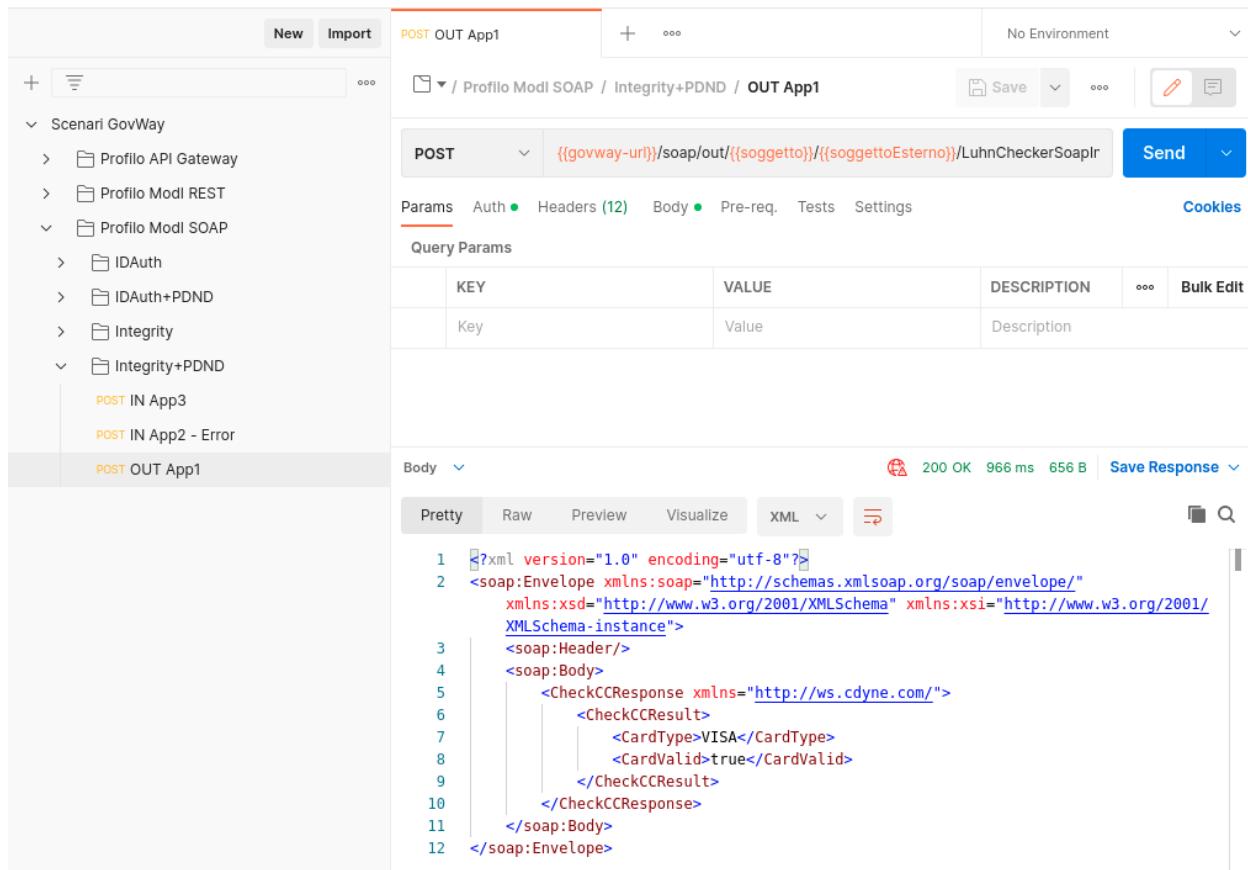


Fig. 3.168: Pattern Integrity+PDND - Fruizione API SOAP, esecuzione da Postman

- Dal dettaglio della richiesta si può visualizzare il messaggio che è stato inviato all'erogatore, come in Fig. 3.169. Come si nota, il messaggio SOAP contiene nell'header WS-Security sia il token di sicurezza (elemento «BinarySecurityToken») sia l'elemento «WSAddressing - To» e il digest del payload (elemento «DigestValue») prodotti dal fruitore con la relativa firma digitale (elemento «SignatureValue») come previsto dal pattern «INTEGRITY_SOAP_01».

Messaggio

```

1  <?xml version="1.0" encoding="UTF-8"?><soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
2    <soapenv:Header>
3      <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" valueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd#BinarySecurityToken">
4        <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd#BinarySecurityToken">MIIE9zCAE+aWIBAgICAPwQ0Y3KoZIhvNAQELBQANjEUMakGAIUEBhMCa0x0EzARBnVBAoMCndvdhEs5VcmcxJjAQBgNVBAMCUdvd1dheSB0DQTAEFw0yMjEwMTKwNzU1NDNaFw0zNzEwMTUwNzU1NDNaMEExAR87a03637e47</wsse:BinarySecurityToken>
5        <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="SIG-0f5d7334-9ad3-42f3-894b-4aba37b2534">
6          <ds:SignedInfo>
7            <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soapenv">
8              <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="soapenv"/>
9            </ds:CanonicalizationMethod>
10           <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
11           <ds:Reference URI="#TS-778700f8-c9d0-4d6c-bfa6-2361c9357a60">
12             <ds:Transforms>
13               <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="wsse soapenv">
14                 <ec:InclusiveNamespaces xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#" PrefixList="wsse soapenv"/>
15               </ds:Transform>
16             </ds:Transforms>
17             <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig#sha256"/>
18             <ds:DigestValue>0gcBkQtbguV2HgY3OK5v05063/3Gmndy73pIHCv18o=</ds:DigestValue>
19           </ds:Reference>
20           <ds:Reference URI="#id-1dcc0908-0d0b-4dd3-bd05-bf1a80722505">
21             <ds:Transforms>

```

Fig. 3.169: Messaggio inviato dal fruitore

Conformità ai requisiti ModI

La verifica dei requisiti ModI per questo scenario non differisce da quanto già descritto in *Esecuzione*.

Configurazione

Nota: Per operare con la govwayConsole in modo conforme a quanto previsto dalla specifica del Modello di Interoperabilità si deve attivare, nella testata dell'interfaccia, il Profilo di Interoperabilità «ModI». Si suggerisce inoltre di selezionare il soggetto “Ente” per visualizzare solamente le configurazioni di interesse allo scenario e nascondere le configurazioni «di servizio» necessarie ad implementare la controparte.



Fig. 3.170: Profilo ModI della govwayConsole

La configurazione dello scenario è del tutto analogo a quello descritto nello scenario *Configurazione* con la sola eccezione del pattern di sicurezza aggiuntivo utilizzato in questo scenario: «INTEGRITY_SOAP_01».

Registrazione API

Viene registrata l'API «CreditCardVerificationIntegrityPDND» con il relativo descrittore WSDL. Vengono selezionati i pattern «ID_AUTH_CHANNEL_01» (sicurezza canale) e «INTEGRITY_SOAP_01» con ID_AUTH_SOAP_01 (sicurezza messaggio) nella sezione «ModI» (Fig. 3.171).

Fruizione

Nella fruizione SOAP “LuhnCheckerSoapIntegrityPDND”, relativa all'API precedentemente inserita, vanno indicati i dati specifici nella sezione «ModI Richiesta» (Fig. 3.83) necessari a generare l'header WS-Security previsto dal pattern

API > CreditCardVerificationIntegrityPDND v1 > **Profilo Interoperabilità**

Profilo Interoperabilità

Note: (*) Campi obbligatori

ModI

Sicurezza Canale

Pattern ▼
Direct Trust Transport-Level Security

Sicurezza Messaggio

Pattern ▼
Integrità payload del messaggio

Applicabilità ▼

Digest Richiesta Non ripudiabilità della trasmissione (i)

Informazioni Utente Dati dell'utente che effettua la richiesta (i)

Fig. 3.171: Configurazione Pattern ModI «INTEGRITY_SOAP_01 con ID_AUTH_SOAP_01» sulla API SOAP

«INTEGRITY_SOAP_01». In particolare è possibile specificare l'audience atteso dall'erogatore e il tempo di validità del token.

ModI - Richiesta

Sicurezza Messaggio	
Algoritmo	RSA-SHA-256
Forma Canonica XML	Exclusive XML Canonicalization 1.0
Riferimento X.509	Binary Security Token
Certificate Chain	<input type="checkbox"/>
Time to Live (secondi) *	60
Indica la validità temporale, in secondi, a partire dalla data di creazione del security token	
WSAddressing To	luhnCheckerSoap.enteEsterno.govway.org 
Indica a chi è riferito il security token; se non viene fornito un valore verrà utilizzata la url del connettore	

Fig. 3.172: Configurazione richiesta della fruizione

La sezione «ModI Risposta» definisce i criteri per la validazione dei messaggi di risposta (Fig. 3.84).

Modi - Risposta

Sicurezza Messaggio

TrustStore Certificati	<input type="text" value="Default"/>
Time to Live	<input type="text" value="Default"/>
Verifica WSAddressing To	<input type="checkbox"/> La verifica utilizza, se configurato, il valore indicato di seguito altrimenti quello configurato nell'applicativo mittente <input type="text"/> (i)

Fig. 3.173: Configurazione risposta della fruizione

CAPITOLO 4

Monitoraggio

In questa sezione descriviamo alcuni tipici scenari di impiego delle funzionalità di monitoraggio offerte da Govway. Il monitoraggio consente di tenere sotto controllo il traffico gestito dal gateway al fine di verificare il regolare funzionamento dei servizi, individuare situazioni anomale ed avviare l'indagine diagnostica.

Per meglio descrivere le attività tipiche della fase di monitoraggio, supponiamo di intervenire nella fase successiva all'esecuzione dei passi dello scenario «Erogazione SPID» (*Erogazione OAuth*).

La console govwayMonitor, nella sezione Monitoraggio, prevede la consultazione del traffico gestito nelle modalità «Storico» e «Live». Ciascuna di queste sezioni mostra l'elenco delle transazioni, in ordine cronologico decrescente, che soddisfano i criteri di filtro impostati ([Fig. 4.1](#)).

Le transazioni riportate nell'elenco riportano i dati per l'identificazione delle stesse, con evidenza dell'esito riportato.

4.1 Transazione in errore

Se apriamo il dettaglio della transazione con esito errore, relativa all'invocazione della «POST /pet» senza token, vediamo le informazioni di [Fig. 4.2](#).

Il dettaglio della transazione:

- Il riquadro «Informazioni Generali» riepiloga i principali dati identificativi della transazione. In questo riquadro è mostrato l'esito, in questo caso negativo. Tramite il link apposito si possono visualizzare i messaggi diagnostici, utili all'identificazione del problema occorso ([Fig. 4.3](#)).
- I riquadri «Dettagli Richiesta» e «Dettagli Risposta» forniscono informazioni specifiche relative al messaggio di richiesta e a quello di risposta. In questo caso, ad esempio, è possibile visualizzare il messaggio di fault inviato al client in risposta ([Fig. 4.4](#)).

Transazioni > Ricerca Base			
Ricerca Base			
Lista Transazioni: record [1 - 6]			
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:23:09, Risorsa API Rest: GET /pet/{petId}	719 ms	HTTP 200	<input type="checkbox"/>
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:22:39, Risorsa API Rest: POST /pet	722 ms	HTTP 200	<input type="checkbox"/>
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:21:43, Risorsa API Rest: POST /pet	66 ms	Gestione Token 401	<input type="checkbox"/>
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:21:21, Risorsa API Rest: POST /pet	93 ms	Token non Presente 401	<input type="checkbox"/>
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:20:19, Risorsa API Rest: GET /pet/findByStatus	783 ms	HTTP 200	<input type="checkbox"/>
PetStore@Ente v1			<input type="checkbox"/>
Data: 2020-11-16 16:19:33, Risorsa API Rest: GET /pet/findByStatus	599 ms	HTTP 302	<input type="checkbox"/>

Fig. 4.1: Elenco delle transazioni

Visualizza Transazioni (Live) > **Dettaglio Transazione**

Dettagli Transazione

Informazioni Generali

Tipologia	Erogazione (API Gateway)
Erogatore	Test
API	PetStore v1
Azione	POST_pet
Esito	Gestione Token Fallita
Diagnostic	Visualizza Esporta

Dettagli Richiesta

Data Ingresso	2019-09-04 16:24:05.876 CEST
Bytes Ingresso	n.d.
Bytes Uscita	n.d.

Dettagli Risposta

Data Uscita	2019-09-04 16:24:05.878 CEST
Bytes Ingresso	143 B
Bytes Uscita	143 B
Fault Uscita	Visualizza

Informazioni Mittente

Metodo HTTP	POST
URL Invocazione	[in] /govway/in/Test/PetStore/v1/pet
Indirizzo Client	127.0.0.1
Codice Risposta Client	400

Informazioni Avanzate

ID Transazione	5fcf5ee0-7588-4313-bcdd-3a7840289aa7
Dominio (ID)	domain/gw/GovWay
Dominio (Soggetto)	GovWay
Latenza Totale	2 ms
Latenza Servizio	N.D.
Latenza Gateway	2 ms
Porta Inbound	__gw_Test/PetStore/v1__Specific1
Applicativo Erogatore	gw_Test/gw_PetStore/v1

Visualizza Transazioni (Live) > Dettagli Transazione > Messaggi Diagnostici			
Lista Diagnostici: record [1 - 6] su 6			
Data	Severità	Funzione	Messaggio
2019-09-04 16:24:05.875	infoIntegration	RicezioneBuste	Ricevuta richiesta applicativa
2019-09-04 16:24:05.877	infoIntegration	RicezioneBuste	Gestione Token [KeyCloak] (Validazione JWT) in corso ...
2019-09-04 16:24:05.877	errorIntegration	RicezioneBuste	<p>Non è stato riscontrato un token nella posizione [RFC 6750 - Bearer Token Usage];</p> <p>(Authorization Request Header) Non è stato riscontrato un header http 'Authorization' valorizzato tramite autenticazione 'Bearer ' e contenente un token</p> <p>(URI Query Parameter) Non è stato riscontrata la proprietà della URL 'access_token' contenente il token</p> <p>(Form-Encoded Body Parameter) Non è stato riscontrata la presenza di un contenuto 'Form-Encoded'</p>
2019-09-04 16:24:05.878	errorIntegration	RicezioneBuste	Gestione Token [KeyCloak] (Validazione JWT) fallita
2019-09-04 16:24:05.878	errorProtocol	RicezioneBuste	Generato messaggio di cooperazione di Errore con identificativo [9419b58e-7693-434f-b1df-fec9e1dda772]
2019-09-04 16:24:05.879	infoIntegration	RicezioneBuste	Risposta ({"type":"https://httpstatuses.com/400","title":"Bad Request","status":400,"detail":"Token non presente","govway_status":"protocol:GOVWAY-1366"}) consegnata al mittente con codice di trasporto: 400

ESPORTA

Fig. 4.3: Messaggi diagnostici della transazione in errore



```

1  {
2   "type" : "https://httpstatuses.com/400",
3   "title" : "Bad Request",
4   "status" : 400,
5   "detail" : "Token non presente",
6   "govway_status" : "protocol:GOVWAY-1366"
7 }

```

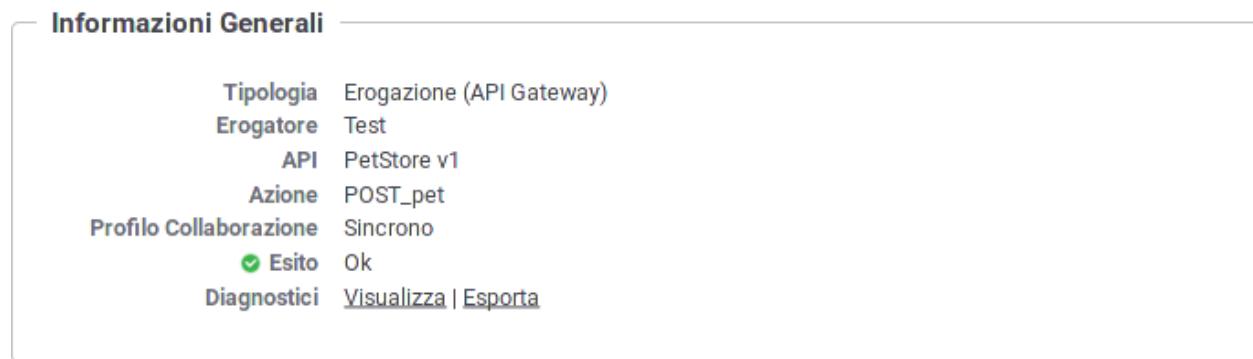
Fig. 4.4: Fault in uscita

- Il riquadro «Informazioni Mittente» fornisce dettagli sulla provenienza della richiesta.
- Il riquadro «Informazioni Avanzate» fornisce dati aggiuntivi riguardo la transazione.

4.2 Transazione con esito corretto

Se apriamo il dettaglio della transazione con esito positivo, relativa all'invocazione della «POST /pet», possiamo ad esempio:

- Visualizzare le informazioni generali con l'esito dell'operazione (Fig. 4.5).



Tipologia	Erogazione (API Gateway)
Erogatore	Test
API	PetStore v1
Azione	POST_pet
Profilo Collaborazione	Sincrono
Esito	Ok
Diagnostici	Visualizza Esporta

Fig. 4.5: Messaggi diagnostici della transazione con esito regolare

- Nel contesto delle informazioni generali si possono visualizzare i messaggi diagnostici con il dettaglio dell'elaborazione regolarmente eseguita (Fig. 4.6).
- Nel contesto delle informazioni mittente in questo caso sarà presente la sezione «Token Info» che consente di visualizzare dati inerenti il token che è stato fornito con la richiesta del mittente. Risultano immediatamente visibili le informazioni principali (issuer, subject, ...), come mostrato in Fig. 4.7.

Visualizza Transazioni (Live) > Dettagli Transazione > Messaggi Diagnostici			
Lista Diagnostici: record [1 - 8] su 8			
Data	Severità	Funzione	Messaggio
2019-09-05 11:32:00.804	infoIntegration	RicezioneBuste	Ricevuta richiesta applicativa
2019-09-05 11:32:00.806	infoIntegration	RicezioneBuste	Gestione Token [KeyCloak] (Validazione JWT) in corso ...
2019-09-05 11:32:00.808	infoIntegration	RicezioneBuste	Gestione Token [KeyCloak] (Validazione JWT) completata con successo
2019-09-05 11:32:01.083	infoProtocol	RicezioneBuste	Ricevuto messaggio di cooperazione con identificativo [222152f4-f8a6-410c-831e-4da92b121f41]
2019-09-05 11:32:01.154	infoProtocol	ConsegnaContenutiApplicativi	Invio Messaggio di cooperazione con identificativo [222152f4-f8a6-410c-831e-4da92b121f41] in corso (location: http://petstore.swagger.io/v2/pet http-method:POST) ...
2019-09-05 11:32:01.521	infoProtocol	ConsegnaContenutiApplicativi	Messaggio applicativo con ID [222152f4-f8a6-410c-831e-4da92b121f41] consegnato al servizio applicativo [gw_Test/gw_PetStore/v1] mediante connettore [http] (location: http://petstore.swagger.io/v2/pet http-method:POST) con codice di trasporto: 200
2019-09-05 11:32:01.524	infoProtocol	RicezioneBuste	Generato messaggio di cooperazione con identificativo [c6991eca-fde0-4065-87a0-bf78410283c8]
2019-09-05 11:32:01.526	infoIntegration	RicezioneBuste	Risposta consegnata al mittente con codice di trasporto: 200

ESPORTA

Fig. 4.6: Messaggi diagnostici della transazione con esito regolare

Informazioni Mittente

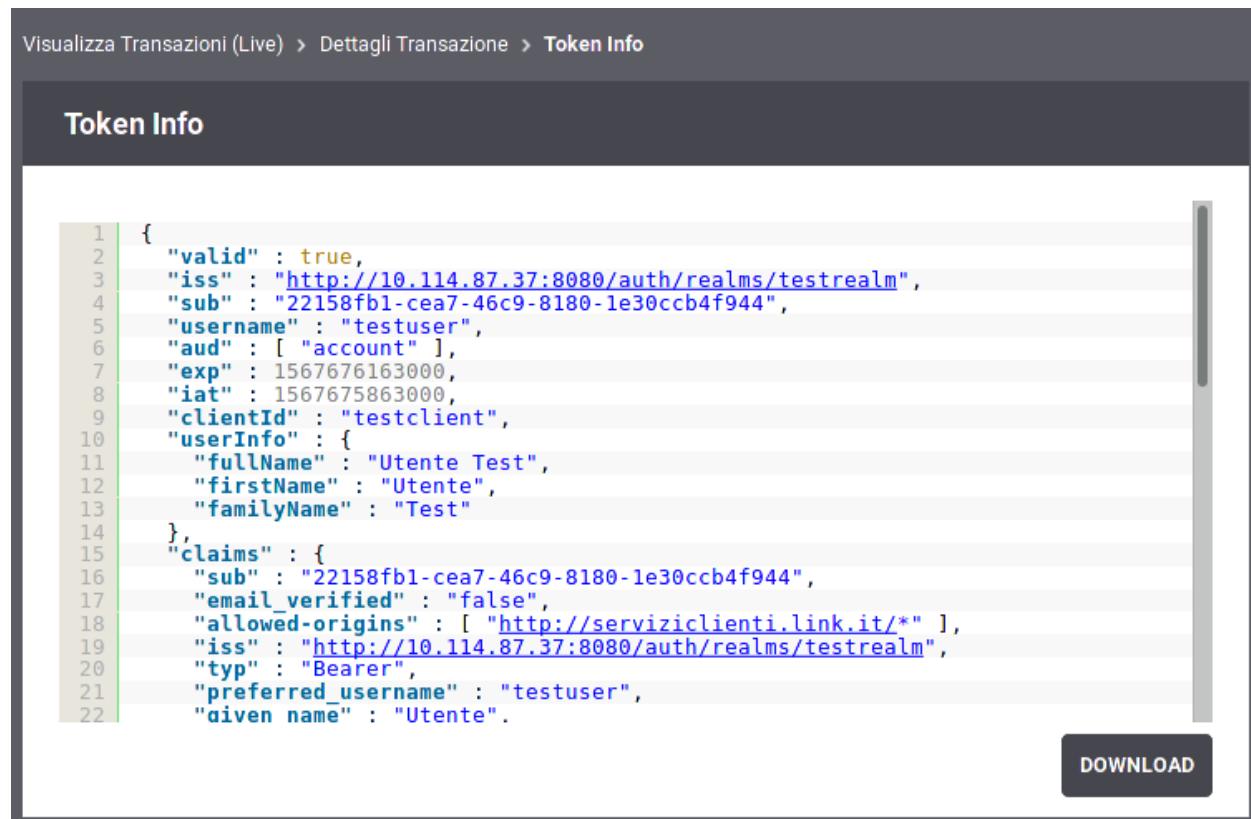
Metodo HTTP	POST
URL Invocazione	[in] /govway/in/Test/PetStore/v1/pet
Indirizzo Client	127.0.0.1
Codice Risposta Client	200

Token Info

Issuer	http://10.114.87.37:8080/auth/realms/testrealm
Client ID	testclient
Subject	22158fb1-cea7-46c9-8180-1e30ccb4f944
Username	testuser
Token Info	Visualizza

Fig. 4.7: Informazioni mittente con presenza del token

- Dalla sezione mittente è possibile aprire una finestra per visualizzare la versione in chiaro del token ricevuto con la richiesta (Fig. 4.8).



Visualizza Transazioni (Live) > Dettagli Transazione > **Token Info**

Token Info

```
1  {
2   "valid" : true,
3   "iss" : "http://10.114.87.37:8080/auth/realms/testrealm",
4   "sub" : "22158fb1-cea7-46c9-8180-1e30ccb4f944",
5   "username" : "testuser",
6   "aud" : [ "account" ],
7   "exp" : 1567676163000,
8   "iat" : 1567675863000,
9   "clientId" : "testclient",
10  "userInfo" : {
11    "fullName" : "Utente Test",
12    "firstName" : "Utente",
13    "familyName" : "Test"
14  },
15  "claims" : {
16    "sub" : "22158fb1-cea7-46c9-8180-1e30ccb4f944",
17    "email_verified" : "false",
18    "allowed-origins" : [ "http://serviziclienti.link.it/*" ],
19    "iss" : "http://10.114.87.37:8080/auth/realms/testrealm",
20    "typ" : "Bearer",
21    "preferred_username" : "testuser",
22    "given_name" : "Utente".
```

DOWNLOAD

Fig. 4.8: Visualizzazione del token