Carte Tokenizzate - Design Review

⚠ WORK IN PROGRESS

Owner (editor)	@ Debora Arena @ Danilo Messina @ Ferro Giuseppe
Team	*CentroStella*
Product(s)	*Centrostella, App IO*
Type	*nuova funzione*
RFC link	Carte Tokenizzate - Design Consultation (RFCs)
Security Review link	*Link alla Security Review*

Glossario

Nella tabella di seguito riportata si riassume la lista dei termini e degli acronimi riportati nel documento in oggetto.

Acronimo	Definizione
BPD	Bonus Pagamenti Digitali.
FA	Fatturazione Automatica
T&C	Terms & Conditions.
CF	Codice Fiscale dell'utente.
PAR	Payment Account Reference. Dato non PCI che identifica univocamente presso i circuiti uno strumento di pagamento.
PAN	Primary Account Number.
HPAN	Hash del PAN (Personal Account Number) di uno strumento di pagamento digitale utilizzato dal cittadino o dall'impresa.
TokenPAN	PAN tecnico utilizzato per i pagamenti con wallet digitali, da intendersi come figlio di un PAN. è associato al medesimo PAR del PAN della carta a cui fa riferimento.
HToken	Hash del Token.
PM	Payment Manager.
TKM	Token Manager: nuovo modulo applicativo da realizzare al fine di poter gestire le carte tokenizzate.
N.A.	Not applicable.

Abstract

Il problema

Allo stato dell'arte l'APP IO non è in grado di enrollare i TokenPAN collegati ad una carta fisica (identificata dal PAN padre), a differenza degli Issuer che invece sono in grado di inviare oltre al PAN anche i token ed il PAR (come già previsto nella soluzione quick&dirty).

Si rende dunque necessario un processo integrativo che assicuri una gestione complessiva dei token associati agli strumenti di pagamento.

La soluzione

La soluzione target serve per sopperire al problema sopra evidenziato, attraverso l'inclusione di un nuovo parametro, ovvero il PAR (Payment Account Reference). Quest'ultimo è definibile come il *collettore* capace di associare i TokenPAN al PAN padre della carta fisica grazie all'associazione unica dei immutabile tra PAN e PAR e Token e PAR.

Si precisa che per gli Issuer le due soluzioni saranno inizialmente complementari.

Fasi principali della soluzione:

- Consenso esplicito del cittadino, tramite APP IO o dai touch point dell'Issuer, all'"utilizzo delle carte tokenizzate". Tale passaggio è un prerequisito necessario per il successivo enrollment delle carte tokenizzate e serve inoltre per:
    1. evitare di sollecitare inutilmente le controparti interessate, ovvero i circuiti: si stimano almeno 4 Milioni di chiamate al giorno per gli strumenti di pagaemnto sprovvisti di PAR.
    2. dare esplicita evidenza al cittadino delle sue azioni.
- Recupero delle associazioni esistenti tra la carta padre a token figli tramite il PAR utilizzando i servizi esposti dal circuiti internazionali.
- Allineamento della base dati del sistema BPD al fine di includere le nuove associazioni HPAN/PAR/HTokenPAN.

La soluzione in oggetto prevede la seguente **strategia:**

- realizzare un modulo applicativo **"Token Manager"** nell'ecosistema del Payment Manager, stand alone deployato con le medesime modalità dei microservizi Centro Stella (BPD) sulla subscription Azure PCIE di SIA, per minimizzare gli impatti sulle altre componenti e centralizzare le integrazioni con i 3 circuiti.
- Il nuovo microservizio (pm-ms-token manager- aka "TKM") utilizzerà un DB cifrato con chiave proprietaria in cui storare le associazioni PAR e relativi HPAN (siano essi PAN o tokenPAN) ed è quindi un componente PCI.
- Il nuovo microservizio TKM esporrà tramite API rest (attraverso APIM) i servizi per sfruttare nativamente la cache esterna Redis.
- Con schedulazione giornaliera, il TKM recupera la lista dei BIN range dai 3 circuiti e li memorizza su una tabella del proprio DB che contiene come minime info:
    - circuito
    - BIN range (num)
    - dedicato a token (booleano)
    - BIN issuer (num)
    - Issuer (string)
    - timestamp inserimento
    - user inserimento
    - timestamp update
    - user update

Matrice degli Impatti e responsabilità

	Moduli/UC	APP IO	Issuer (non incluso in Q&D)	PM	TKM	Centro Stella	Acquirer
1	Consenso da parte dell'utente ad utilizzare le carte tokenizzate	***esposizione in APP della nuova funzionalità***	***esposizione in APP della nuova funzionalità***	*n.a.*	*(OWNER PRINCIPALE)* *API esposta da TKM*	*eventuale impatto sull' APIM*	*n.a.*
2	Recesso dell'utente dall'utilizzo delle carte tokenizzate	***esposizione in APP della nuova funzionalità***	***esposizione in APP della nuova funzionalità***	*n.a.*	*(OWNER PRINCIPALE)* *API esposta da TKM*	*eventuale impatto sull' APIM*	*n.a.*

3	**Recupero dello stato di accettazione/disattivazione del consenso da parte dell'utente ad utilizzare le carte tokenizzate**	***esposizione in APP della nuova funzionalità***	***esposizione in APP della nuova funzionalità***	*n.a.*	*(OWNER PRINCIPALE)* *API esposta da TKM*	*eventuale impatto sull' APIM & integrazione alla nuova funzionalità*	*n.a.*
4	**Processo di allineamento del TKM di tutte carte presenti sul PM**	*n.a.*	*n.a.*	*scrittura su coda dei nuovi wallet*	*(OWNER PRINCIPALE)* *lettura da una coda condivisa con il PM per riceverei nuovi wallet per i quali è necessario recuperare il PAR*	*n.a.*	*n.a.*
5	**Cancellazione di uno strumento di pagamento ed allineamento del TKM**	*n.a.*	*Richiesta di cancellazione di una carta*	*scrittura su coda condivisa con TKM dei wallet cancellati*	*lettura da una coda condivisa con il PM per ricevere la lista dei wallet da cancellare*	*TBD (per integrazione UC di cancellazione da Issuer)*	*n.a.*
6	**Processi di recupero ed associazione delle carte padre con il parametro PAR (HPAN-PAR)**	*n.a.*	*n.a.*	*n.a.*	*(OWNER PRINCIPALE)* *1. meccanismo batch di recupero PAR per wallet sprovvisti di PAR e per i quali l'utente ha dato il consenso.* *2. comunicazione al CS delle nuove associazioni*	*recupero dal TKM nuove associazioni ed aggiornamento base dati solo per carte già enrollate*	*n.a.*

7	**Processi di recupero dai circuiti dei Bin Range**	*n.a.*	*n.a.*	*n.a.*	*(OWNER PRINCIPALE)* *il TKM invoca i servizi esposti dai circuiti per recuperare /aggiornare la lista dei BIN abilitati alla tokenizzazione.*	*n.a.*	*n.a.*
8	**Integrazione Batch Acquirer**	*n.a.*	*n.a.*	*n.a.*	*-recupero lista bin range , generazione flusso e caricamento su sfTP(TBD)*	*(OWNER PRINCIPALE)* *nuove logiche da integrare lato Batch Acquirer per le 3 liste*	*n.a.*
9	**Inserimento TokenPAN su TKM e recupero PAR**	*n.a.*	*n.a.*	*n.a*	*(OWNER PRINCIPALE)* *il TKM legge la lista Token e recupera il PAR dai circuiti e successivamente comunica le nuove associazioni al CS*	*Impatto sul batch acquirer per elaborazione lista bin range /token e creazione flusso da inviare al TKM + [recupero dal TKM nuove associazioni ed aggiornamento base dati solo per carte già enrollate (API patch già presente nelle Q&D)]*	*generazione ed invio lista token appartenenti ai bin range*

10	**Integrazione Acquirer**	*n.a.*	*n.a.*	*n.a.*	*n.a.*	*flusso transazioni: aggiornamento tracciati*	*(OWNER PRINCIPALE)* *modifica applicativa da parte degli acquirer come aggiunta di una nuova entità (PAR) +integrazione nuove logiche per creazione lista tokenPAN*
11	**Flusso delle transazioni: adattamento delle logiche di match e salvataggio nuove associazioni**	*n.a.*	*n.a.*	*n.a.*	*salvataggio nuove associazione (HPAN /Htoken e PAR)*	*(OWNER PRINCIPALE)* *Centro Stella: nuove logiche di match & comunicazione al TKM delle nuove associazioni e salvataggio di nuovi parametri (Htoken o PAR)*	*n.a.*

Assumption

ID	UC	Descrizione assumption
1	N.A	L'associazione fra PAN e PAR è univoca ed immutabile.
2	N.A	L'associazione fra TokenPAN e PAR è univoca ed immutabile.
3	N.A	I tokenPAN sono PAN con bin range dedicati.
4	N.A	I 3 principali circuiti (Mastercard, Visa, Amex) espongono servizi tramite API per: • scaricare la lista di BIN dedicati ai relativi tokenPAN • recuperare PAR dato tokenPAN • recuperare PAR dato PAN.
5	N.A	Si prevede una modifica sul PM in modo che salvi per ciascun wallet il valore dell'HPAN (utilizzando il SALT).
6	N.A	I 3 circuiti si comportano in modo differente e per VISA e Amex il PAR viene generato solo dopo la generazione del primo tokenPAN, mentre per Mastercard il PAR è sempre disponibile.
7	TK1 TK2	Il processo di richiesta del consenso/recesso da parte del cittadino è lo stesso sia per APP IO che Issuer. Si prevede pertanto un impatto sul processo di onboarding da Issuer, non incluso attualmente nelle Q&D.

8	TK1 TK2	L’informazione in merito il consenso/recesso sarà storicizzata solo sul TKM e la stessa verrà propagata verso sistemi interessati, tramite l’api di get status citizen tokenizzate.
9	TK4	Il PM non salverà il PAR e HTokenPAN sulla propria base dati.
10	TK5	La cancellazione di un metodo di pagamento (HPAN) comporta la disattivazione dello stesso e di eventuali token ad asso associati
11	TK6 TK9	In caso di enrollment/aggiornamento Token tramite chiamata API da H/M Banking Issuer, CentroStella veicolerà l’informazione verso il PM per la procedura di hashing e allineamento del TKM, ma non verrà salvato nessun dato relativo ai token nel database di BPD. Il salvataggio verrà effettuato solo a seguito della ricezione dell’informazione sui Token (e Hpan e PAR associati) dal TKM. Si precisa quindi che in fase di enrollment carta verrà memorizzato solo l'HPAN ma non i token associati.

Requisiti

Funzionali: casi d’uso

- TK1: Consenso da parte dell'utente ad utilizzare le carte tokenizzate
- TK2: Dissenso dell’utente dall’utilizzo delle carte tokenizzate
- TK3: Recupero dello stato di accettazione/disattivazione del consenso da parte dell'utente ad utilizzare le carte tokenizzate
- TK4: Processo di allineamento del TKM di tutte carte presenti sul PM
- TK5: Cancellazione di uno strumento di pagamento ed allineamento del TKM
- TK6: Processi di recupero ed associazione delle carte padre con il parametro PAR (HPAN-PAR)
- TK7: Processi di recupero dai circuiti dei Bin Range
- TK8: Integrazione Batch Acquirer
- TK9: Inserimento TokenPAN sul TKM e recupero PAR
- TK10: Integrazione Acquirer
- TK11: Flusso delle transazioni: adattamento delle logiche di match e salvataggio nuove associazioni

Non funzionali

Indicare i vincoli ambientali/prestazionali/normativi rilevanti ai fini delle scelte prese. Ad es. numero di transazioni per secondo (TPS) da sostenere, ridondanza geografica del servizio, obbligatorietà dell’uso di certi servizi o middleware, etc.

GDPR

Indicare se GDPR è stato considerato. Se no, perché. Se si, indicare quali servizi/API sono messe a disposizione per i casi d’uso richiesti dal GDPR.

Build vs. buy?

Sono state esaminate e scartate soluzioni pronte “as-is” ed “off the shelf”? Se si, quali e per quali ragioni scartate?

Design di alto livello

Fai clic qui per espandere...

Infra	*Cloud*
Cloud	*Azure*
Cloud: regione	*WestEurope*
Cloud: Account	*SIA - sub : U87PagoPa*
On-Prem	*n.a.*

On-Prem: regione	*n.a.*

Vista statica delle componenti

Component Diagram: carte tokenizzate target - diagrams.ne

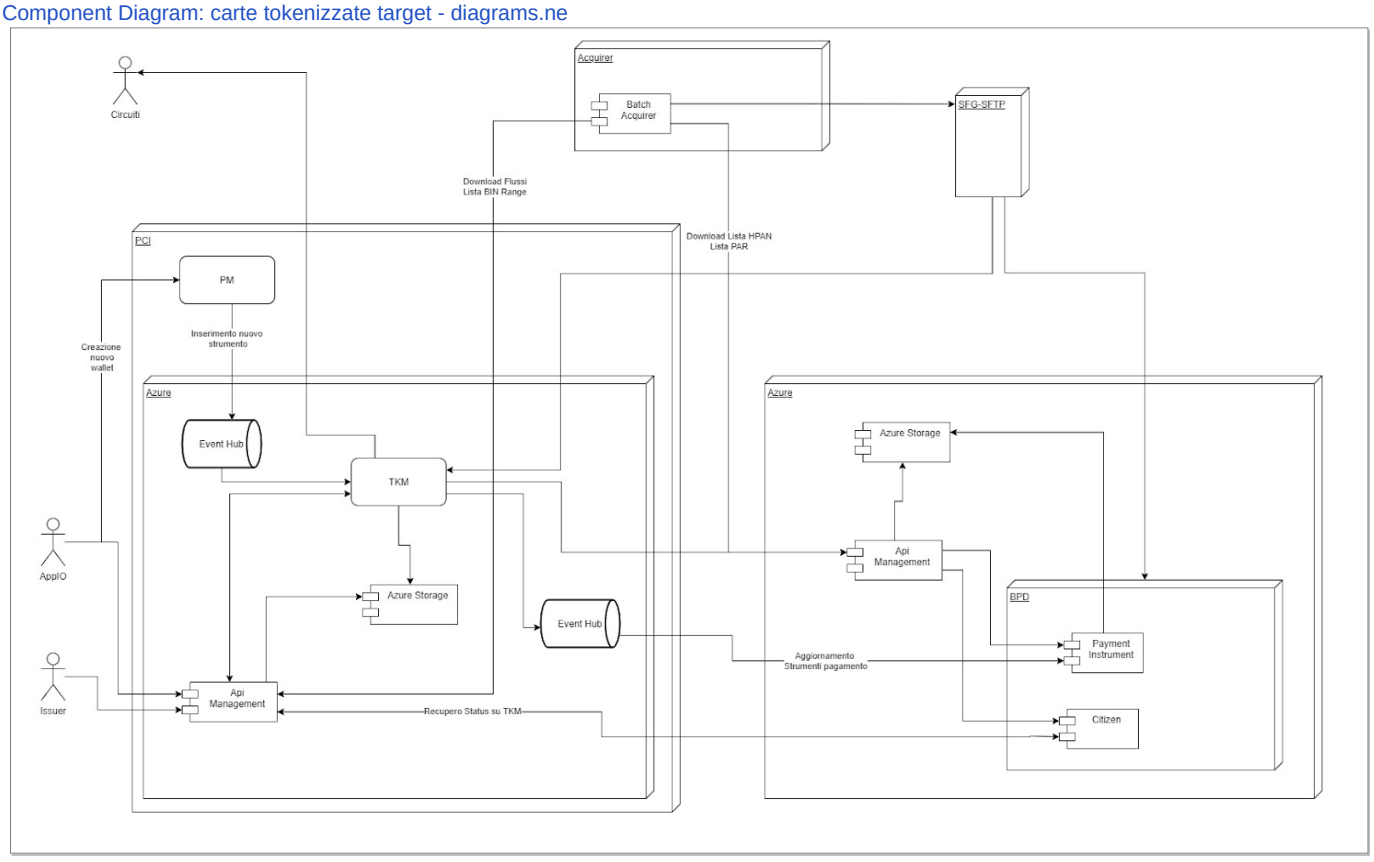
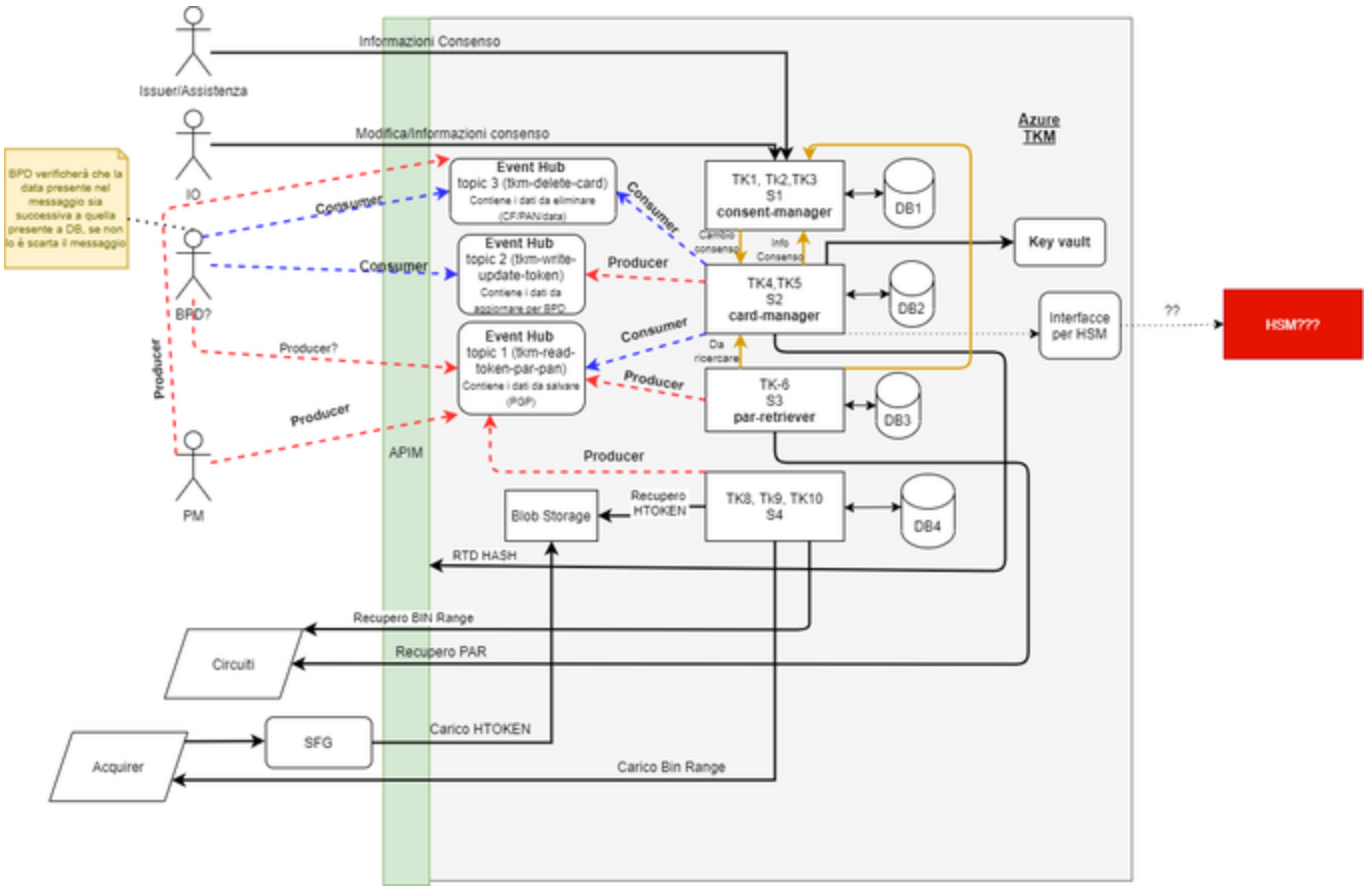


Diagramma microservizi TKM.drawio



Tkm Repository:

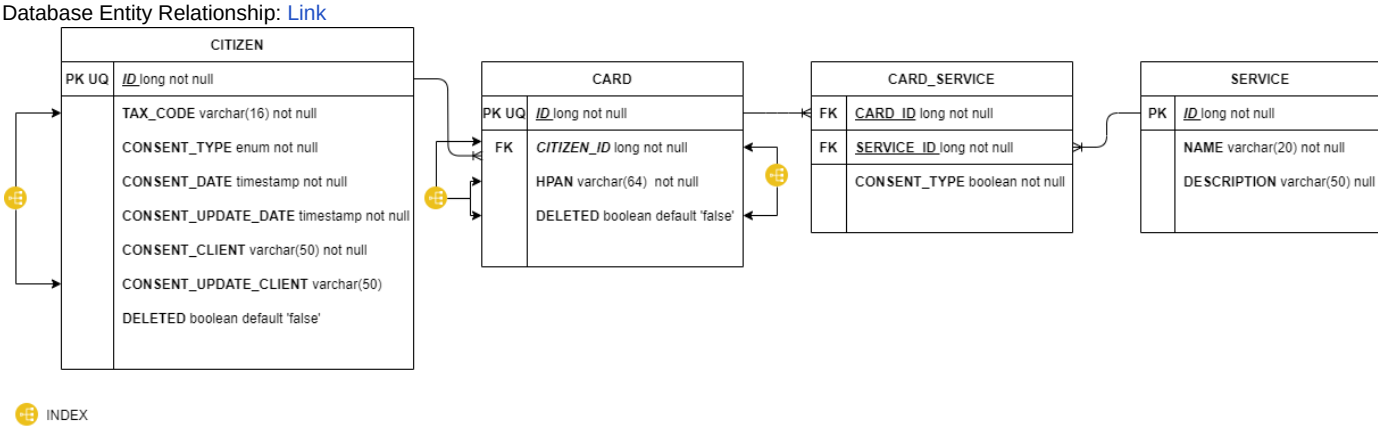
- Tkm Repository PagoPA Open-Source

Tkm-ms-consent-manager(TK1, TK2) Technical Details

OpenAPI Specification (internal/issuer/IO):

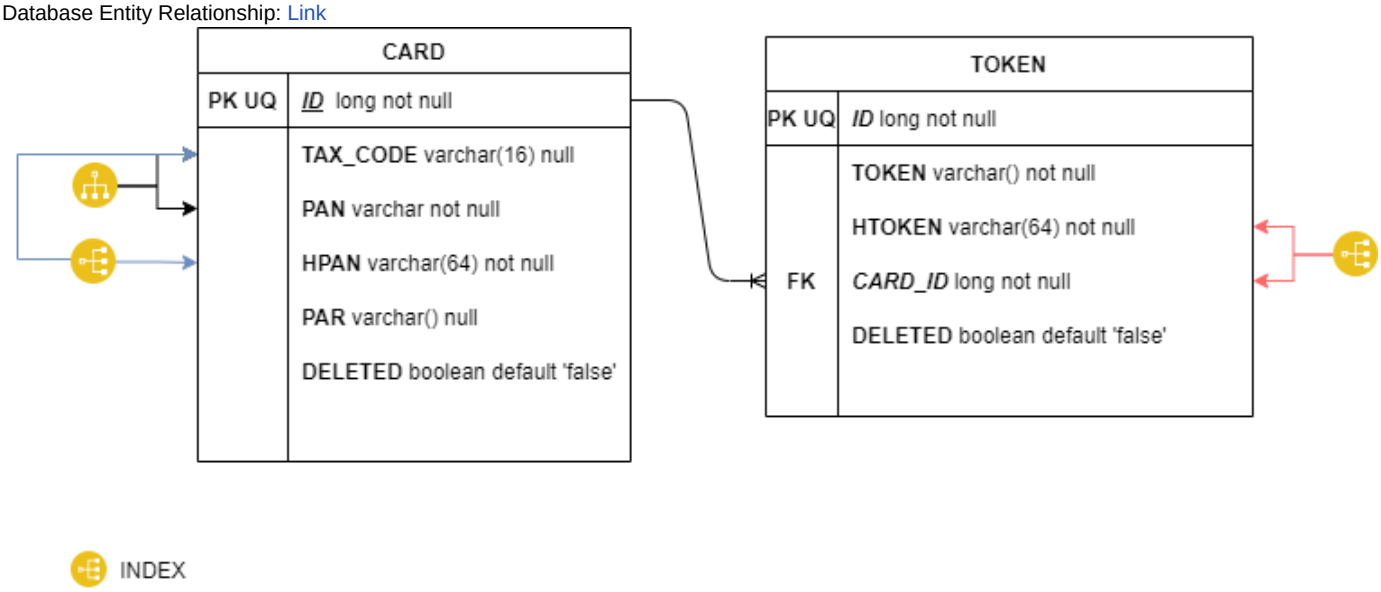
- SIT: tkm-ms-consent-manager/openapispec at release/sit · pagopa/tkm-ms-consent-manager · GitHub
- UAT:
- PROD:

Database Entity Relationship: Link



Tkm-ms-card-manager Technical Details

Database Entity Relationship: Link



Event Hub Topic 1 Technical Details

All'interno del topic **tkm-read-token-par-pan**, nell'event hub namespace **ddstkmhub,** saranno scambiati messaggi cifrati tramite protocollo crittografico PGP, la cui chiave è generata e distribuita (quella pubblica) dal TKM. La chiave privata è mantenuta all'interno del key vault.

Il topic è stato configurato nel seguente modo:

- 6 partizioni
- 3 giorni di retention

I messaggi (in chiaro) hanno una struttura JSON così definita:

```
ReadQueue:
  type: object
  required:
    - taxCode
    - pan or par
    - circuit
  properties:
    taxCode:
      type: string
      minimum: 16
      maxLength: 16
    pan:
      type: string
    hpan:
      type: string
    par:
      type: string
    circuit:
      type: string
      enum:
        - Visa
        - Mastercard
        - Amex
    tokens:
      type: array
      items:
        $ref: '#/components/schemas/Tokens'

Tokens:
  type: object
  properties:
    token:
      type: string
    htoken:
      type: string
```

Event Hub Topic 2 Technical Details

All’interno del topic **tkm-write-update-token**, nell'event hub namespace **ddstkmhub,** saranno scambiati messaggi in chiaro non contenenti dati sensibili.

Il topic è stato configurato nel seguente modo:

- 6 partizioni
- 3 giorni di retention
- Group Id: bpd-client (read-only)

I messaggi hanno una struttura JSON così definita:

```
WriteQueue:
  type: object
  required:
    - taxCode
    - hpan
  properties:
    taxCode:
      type: string
      minimum: 16
      maxLength: 16
    timestamp:
      type: string
      format: datetime
      pattern: "yyyy-MM-dd HH:mm:ss:SSSS"
      example: "2021-01-01 00:00:00:0000"
    cards:
      type: array
      items:
        properties:
          hpan:
            type: string
          action:
            type: string
            enum:
              - INSERT_UPDATE
              - REVOKE
          par:
            type: string
            description: Empty if the user revokes the consent or
requests the cancellation
          htokens:
            type: array
              items:
                $ref: '#/components/schemas/Tokens'

Tokens:
  type: object
  properties:
    htoken:
      type: string
    haction:
      type: string
      enum:
        - INSERT_UPDATE
        - DELETE
```

Event Hub Topic 3 Technical Details

All’interno del topic **tkm-delete-card**, nell'event hub namespace **ddstkmhub,** saranno scambiati messaggi in chiaro non contenenti dati sensibili..

Il topic è stato configurato nel seguente modo:

- 3 partizioni
- 2 giorni di retention
- Group Id: bpd-client (read-only)

I messaggi hanno una struttura JSON così definita:

```
DeleteMessage:
  type: object
  required:
    - taxCode
    - hpan
  properties:
    taxCode:
      type: string
      minimum: 16
      maxLength: 16
    hpan:
      type: string
      minimum: 64
      maxLength: 64
    timestamp:
      type: string
      format: datetime
      pattern: "yyyy-MM-dd HH:mm:ss:SSSS"
      example: "2021-01-01 00:00:00:0000"
```

Vista dinamica delle componenti

TK1. Consenso da parte dell'utente ad utilizzare le carte tokenizzate

Il consenso all’utilizzo delle carte tokenizzate va inteso come l’autorizzazione da parte dell’utente a poter recuperare PAR/tokenPAR attraverso processi paralleli innescati dal TKM (invocazione dei circuiti ecc.) rispetto a quelli effettuati direttamente dall’Issuer (enrollement carte tokenizzate o aggiornamento della lista token associata ad una carta padre).

L’utente tramite i canali App IO o dai touch point dell’Issuer, potrà scegliere di attivare l’utilizzo delle proprie carte tokenizzate per i servizi di BPD e/o FA. Al fine di lasciare all’utente un ampio grado di libertà nella scelta, il consenso verrà effettuato su base CF mentre sarà opzionale sul singolo strumento di pagamento.

A tale proposito il modulo TKM esporrà un nuovo servizio condividendo l’APIM e la modalità di autenticazione usata dal Centro Stella, nel quale saranno veicolate informazioni come il CF, tipoServizio (BPD, FA, ecc..). Le informazioni ricevute in input verranno salvate nella tabella *user* del TKM.

Il consenso sarà detenuto unicamente dal TKM, il quale esporrà delle nuove API di *get user status tokenizzate* per veicolare tale informazione verso i sistemi interessati (per esempio: APP IO che invocherà l’api in oggetto).

Il TKM invierà verso il Centro Stella nuove associazioni solo in presenza di consenso da parte dell’utente. In particolare per ogni strumento di pagamento verranno inviati i seguenti valori

- HPAN
- PAR
- data di enrollment del PAR

Prima di salvare tale parco informativo, il Centro Stella verifica se ha già sulla propria base dati il PAR e solo in assenza dello stesso, storicizza le informazioni ricevute in input e assegna al PAR la data di enrollment generata dal TKM. Tale data verrà assegnata di default a tutti i token associati alla carta padre e servirà come data di riferimento in fase di elaborazione delle transazioni. Per maggiori dettagli si rimanda al TK11.

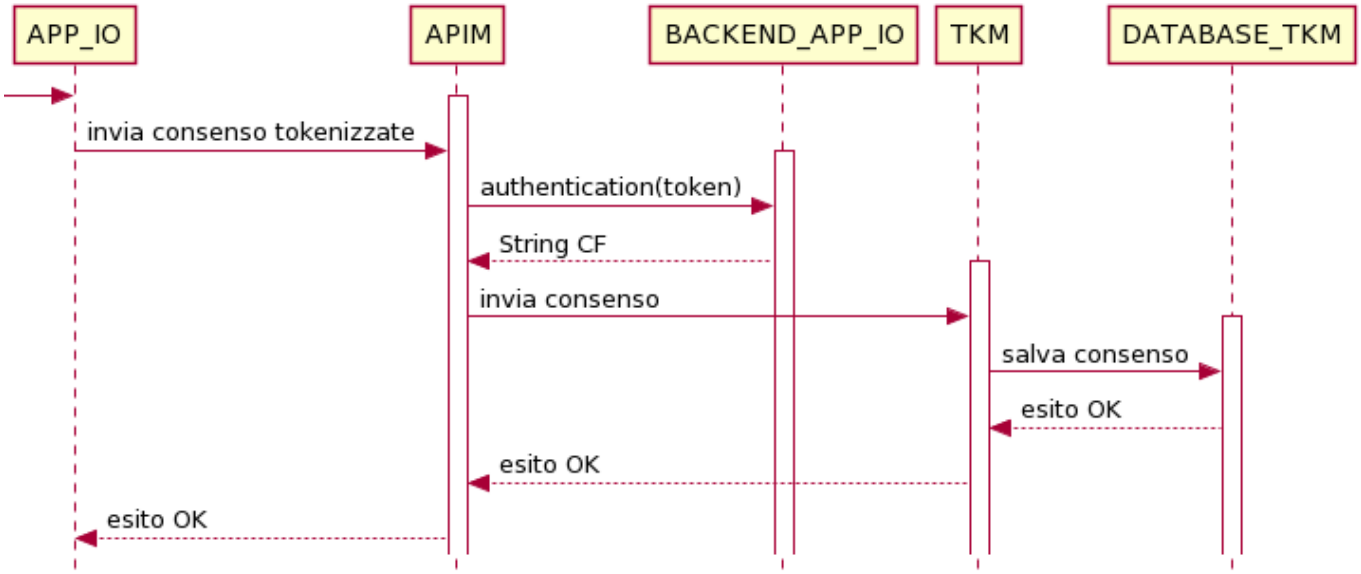
L'invocazione di queste API verso il TKM, scatena la chiamata da parte di TKM utilizzando l'API di patch payment instruments (modificata per ricevere anche il PAR) aggiornerà con i nuovi token BPD e/o FA rispetto alla posizione della carta padre/CF. da questo momento anche HtokenPAN sarà inviato nei flussi verso gli acquirer insieme con HPan

le API sono sincrono e prevedono in input la possibilità di avere opzionale una sola carta (HPAN).

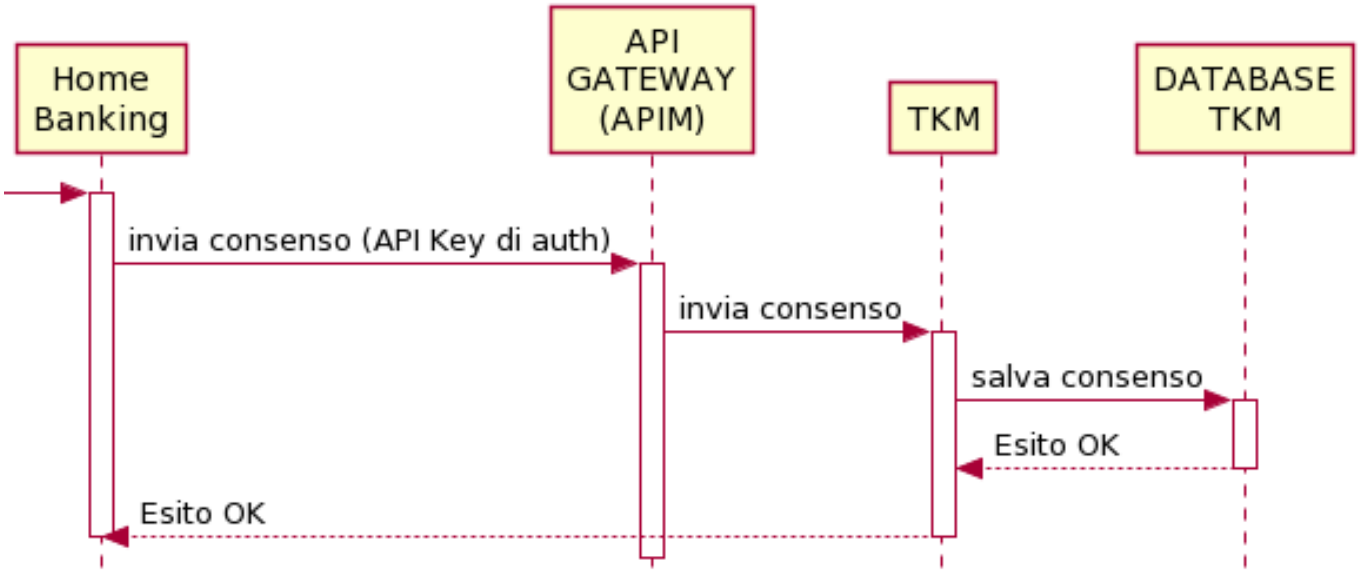
Nel messaggio di attivazione consenso se è presente HPAN allora il consenso è attivato solo sulla singola carta, in assenza di HPAN il consenso è da intendersi su tutte le carte.

In caso di consenso su tutte le carte, i successivi tentativi di attivare o disattivare il consenso su carta singola avranno in risposta un messaggio di errore.

Sequence Diagram: richiesta di consenso da APP IO



Sequence Diagram: richiesta di consenso da Issuer



TK2. Dissenso dell'utente dall'utilizzo delle carte tokenizzate (HP)

Come per il consenso, l'utente potrà scegliere di disattivare l'utilizzo delle proprie carte tokenizzate per i servizi di BPD e/o FA. Tale servizio verrà reso disponibile sia da APP IO che da H/M Banking tramite una API esposta dal TKM. Le informazioni ricevute in input verranno salvate nella tabella *user*:

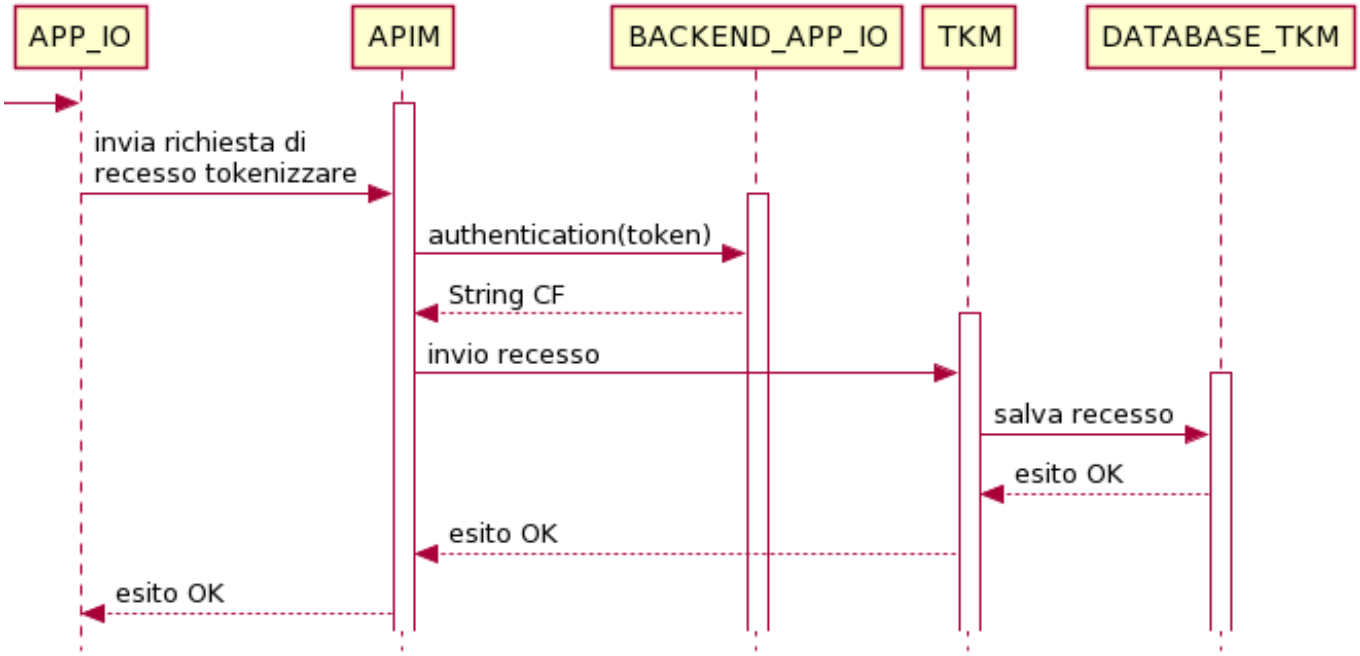
- CF
- deactivation (boolean)
- tipo servizio
- timestamp inserimento
- user inserimento
- timestamp update
- user update

L'informazione sulla disapprovazione sarà detenuta unicamente dal TKM. *In presenza di dissenso da parte dell'utente nel consentire al sistema TKM di recuperare i parametri legati alle carte tokenizzate (PAR/TokenPAN), quest'ultimo dovrà comunicare al Centro Stella "l'aggiornamento" rispetto ad ogni singolo strumento di pagamento. Nello specifico verrà inviato tramite coda un evento contenente:*

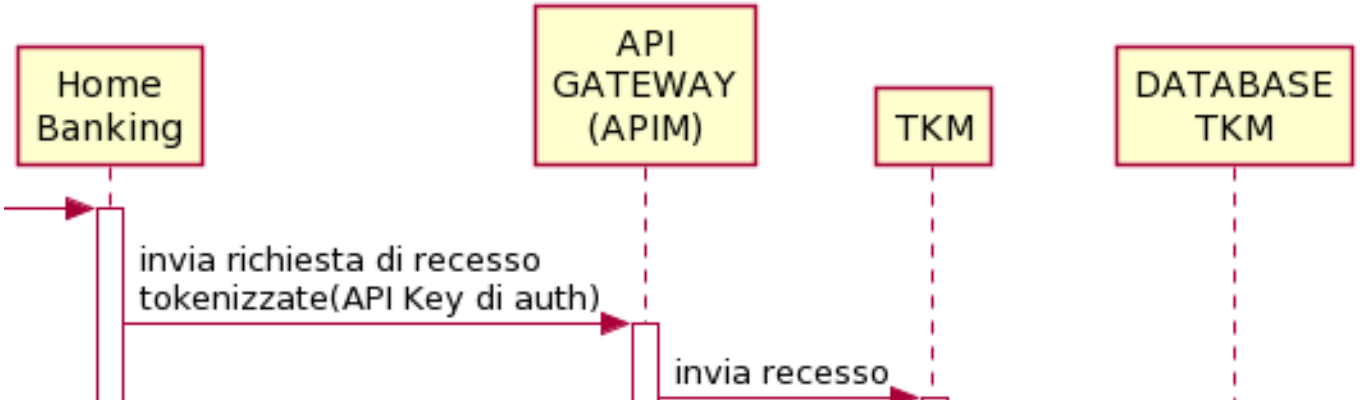
- l'HPAN della carta
- timestamp di disattivazione del PAR. La data di disattivazione verrà associata automaticamente a tutti gli Htoken figli della carta padre e servirà come data di riferimento in fase di elaborazione delle transazioni. Per maggiori dettagli si rimanda al TK11.

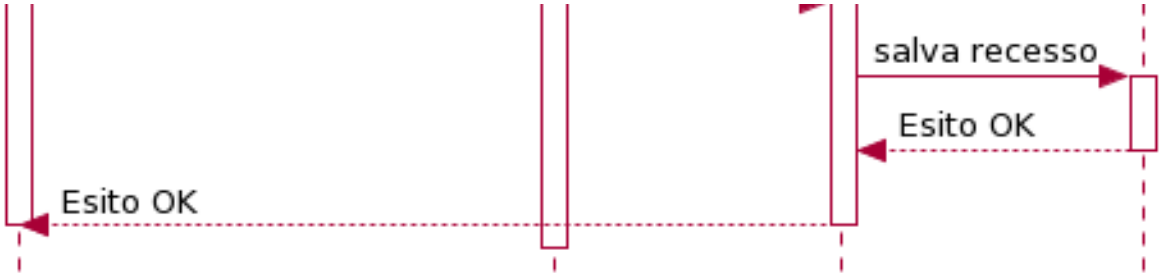
In caso di consenso su tutte le carte, i successivi tentativi di attivare o disattivare il consenso su carta singola avranno in risposta un messaggio di errore.

Sequence Diagram: richiesta di recesso da APP IO



Sequence Diagram: richiesta di recesso da Issuer





TK3. Recupero dello stato di accettazione/disattivazione del consenso da parte dell'utente ad utilizzare le carte tokenizzate

Al fine di ricevere lo stato di accettazione/disattivazione del consenso all'utilizzo delle carte tokenizzate, l'APP IO, H/M Banking potranno invocare l'API di *get user status tokenizzate* esposta dal TKM, esplicitando la tipologia di servizio per il quale sarà necessario recuperare l' informazione. Il set di campi restituito sarà il seguente:

- stato dell' utente (ACTIVE/INACTIVE) sul servizio
- data di accettazione o di recesso utente

Si precisa che il Centro Stella non avrà la necessità di invocare tali API in quanto riceverà sempre valori pre-filtrati rispetto all'informazione del consenso/dissenso da parte del TKM.

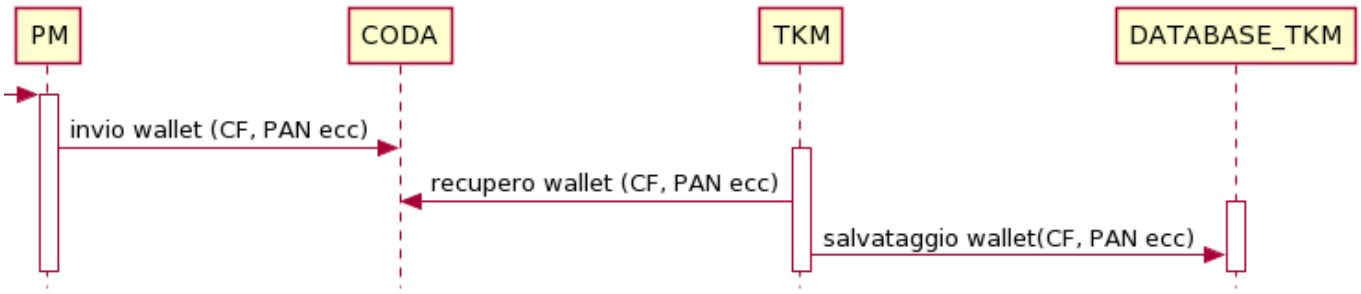
TK4. Processo di allineamento del TKM di tutte carte presenti sul PM

Per tutte le carte censite su PM (compresa la componente RTD), quest'ultimo scriverà in modalità asincrona su una coda, eventi cifrati con chiave pgp pubblica rilasciata da TKM inviando con CF e PAN.

Il modulo TKM come consumer legge la coda e salva le informazioni sul proprio DB sulla tabella *Card:*

- CF
- PAN (cifrato) - PRIMARY KEY
- ultime 4 cifre del PAN
- HPAN (hash del valore con salt richiesto tramite API a PM)
- PAR
- BIN issuer (num)
- timestamp inserimento
- user inserimento
- timestamp update
- user update

Sequence Diagram



TK5. Cancellazione di uno strumento di pagamento ed allineamento del TKM *(in corso)*

Overview generale

Lo Use Case in oggetto definisce il workflow speculare a quello di allineamento del TKM descritto nel paragrafo TK4. In particolare, al fine di allineare il TKM rispetto al set di strumenti di pagamento cancellati sul PM, si rende necessario implementare un processo su una coda kafka che veicoli, in near real time, tali informazioni. Per le carte cancellate il TKM non invocherà più i circuiti al fine di recuperarne il PAR associato.

Con l'integrazione sopra descritta ed in presenza di cancellazione da APP IO/Issuer, il sistema del PM dovrebbe gestire una comunicazione 1:n, comunicando al TKM le carte cancellate ed propagando la medesima informazione verso il sistema RTD, responsabile di disattivare i servizi attivi sullo strumento oggetto di cancellazione. **In ottica di ottimizzare tali processi di allineamento è preferibile definire una comunicazione su coda near real time, adeguando alla stessa anche l'attuale use case "*GEN4c. Disabilitazione carte da PM*" gestito in modalità sincrona attraverso delle API**.

Tuttavia al fine per completare il caso d'uso di cancellazione sarà necessario implementare un ulteriore processo di cancellazione di una carta da *Issuer* attualmente non presente. Lo use case verrà approfondito nel paragrafo successivo.

Sotto processo di cancellazione di uno strumento di pagamento tramite H/M Banking (GAP)

La funzionalità contempla il caso di cancellazione di una carta tramite touch point degli Issuer. Non vigendo una comunicazione diretta tra i sistemi degli Issuer e quello del PM (master sullo stato di creazione/disattivazione dei wallet), a differenza di quanto previsto per l'APP IO, il Centro Stella sarà responsabile di *veicolare* tramite l'APIM le richieste di cancellazione verso il PM.

Come indicato nel paragrafo sopra, una volta finalizzata la cancellazione sul PM, quest'ultimo comunicherà la disattivazione a RTD e la cancellazione al TKM tramite un sistema di code kafka.

Descrizione del processo

a. Cancellazione Wallet sul PM

- All'atto della cancellazione di una carta, l'Issuer invoca un servizio di delete esposto dall'APIM inviando i seguenti dati:
  - PAN della carta cifrato con chiave PGP
  - CF dell'utente
  - token di autenticazione (API Key)
- A seguito dell'autenticazione, l'APIM effettua una redirect sull'API di delete esposta dal PM per comunicare la cancellazione del wallet, inviando:
  - PAN della carta cifrato con chiave PGP
  - CF

Qualora la cancellazione non dovesse andare a buon fine il workflow verrebbe interrotto ed il PM restituirebbe all'APIM un messaggio di errore, il quale verrebbe propagato fino all'Issuer. In caso di esito positivo invece il PM propaga verso RTD la disattivazione e restituisce un esito verso l'APIM.

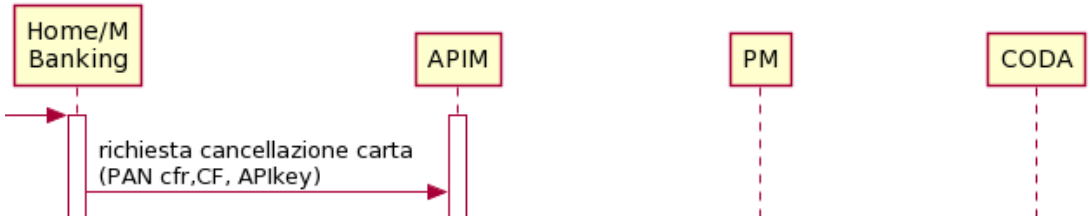
NOTA: il processo di comunicazione tramite coda tra PM e RTD, inerente le carte da disattivare, sarà il medesimo sia in presenza di richiesta di cancellazione da APP IO che da Issuer.

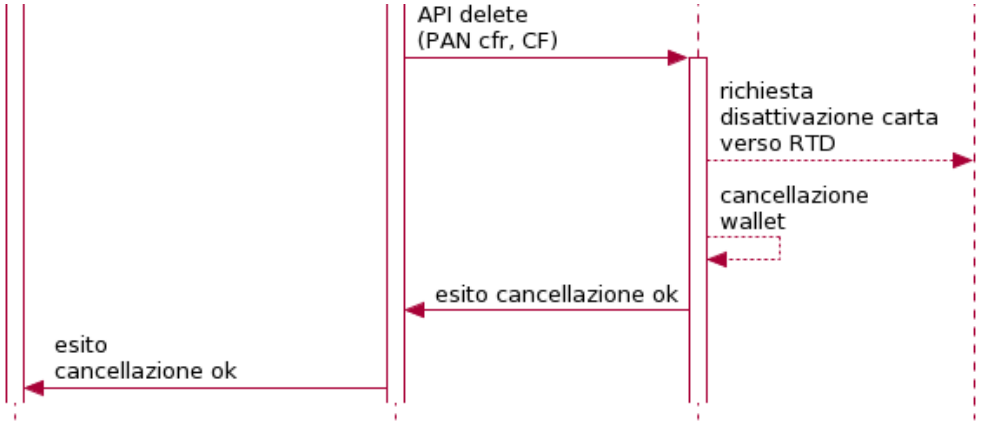
b. Disattivazione carta sul Centro Stella

- A seguito della richiesta di cancellazione da Issuer, il PM veicola la disattivazione verso RTD (i dati forniti non dovranno essere cifrati). Procede dunque con i seguenti step: decripta il PAN cifrato, effettua l'hashing e deposita il parco informativo su una coda letta da RTD, fornendo:
  - HPAN dello strumento di pagamento cancellato
  - CF
  - timestamp di disattivazione
- RTD leggerà dalla coda tutte le volte in cui vi saranno nuovi eventi ed invoca i servizi di *deletePaymentiInstruments* di BPD (in futuro FA):
- BPD verifica se l'HPAN ricevuta da RTD è presente sulla propria base dati interna ed in caso positivo salva i parametri in ingresso, aggiornando contestualmente lo stato dello strumento di pagamento e di tutti gli HToken associati:
  - CF utente
  - HPAN/HToken disattivati
  - aggiornamento dello stato: da Active in Inactive sia per l'HPAN che gli eventuali HToken
  - Timestamp disattivazione staccata dal PM

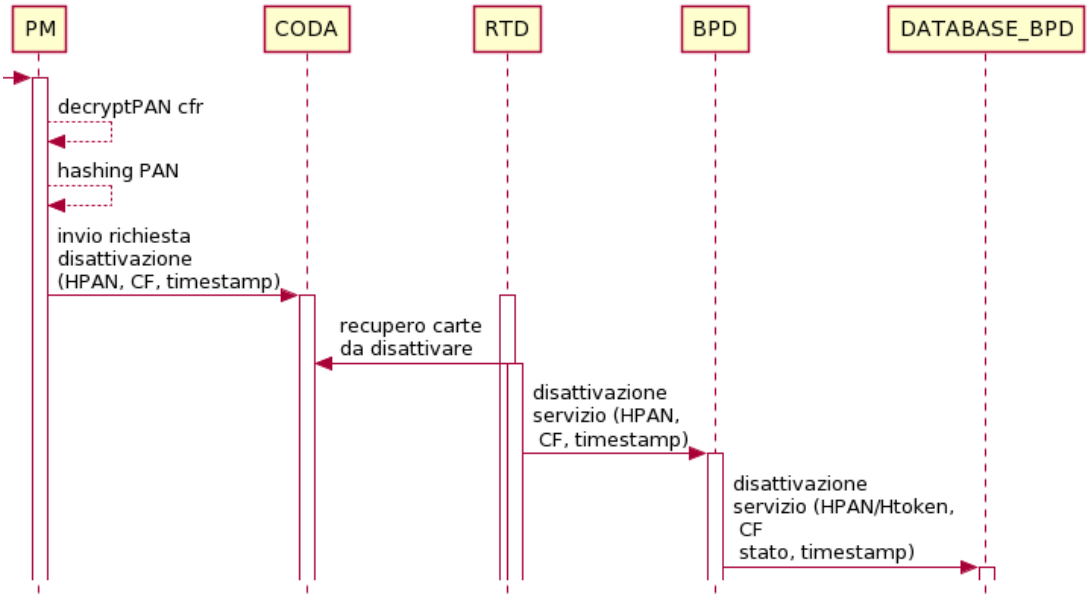
Sequence Diagram

a. Cancellazione Wallet sul PM





b. Disattivazione carta sul Centro Stella



API Rest: cancellazione strumento da H/M Banking

Di seguito si riporta l'API di cancellazione di uno strumento di pagamento richiamata dagli Issuer ed esposta dal PM tramite una redirect dell'API Management. Tale servizio sarà riproposto nel documento di Issuer Interface Agreement.

Path APIM (redirect verso PM): TBD (owner PM)

Path PM: TBD (owner PM)

Method: DELETE

Request Header:

Field	Type	Mandatory	Description
Ocp-Apim-Subscription-Key	Alphanumeric	*YES*	*Subscription key associated with the issuer*
x-request-id	*String*	*NO*	*Request ID, unique identifier determined by initialisation (UUID)*
id			

	Alphanumeric	*YES*	*id of the payment instrument, which corresponds to the pgp encr ypted Primary Account Number (PAN) of the payment method*
user id	*Alphanumeric*	*YES*	*user ID, which corresponds to their tax code*

Response Code: HTTP Response Code 204

Response Header

Field	Type	Mandatory	Description
x-request-id	*String*	*NO*	*Request ID, unique identifier determined by caller or system (UUID)*

HTTP Error Codes

HTTP Response Code	Error code
504	*TIMEOUT*
500	*GENERIC_ERROR*
401	*TOKEN_NOT_VALID*

Descrizione Casi di errore ed eccezioni

- **Timeout:** prevediamo un messaggio anche in caso di timeout nella risposta, in questo caso l'Issuer può ritentare la chiamata al servizio (non sono previste politiche di retry sulla piattaforma BPD, sarà APP IO a gestire il retry nel caso di tentativo fallito).
- **Cancellazione non andata a buon fine da parte del PM:** in questo caso l'intero processo verrà bloccato restituendo al client un messaggio d'errore.
- **Token non valido:** viene considerato come token non valido la risposta incompleta del servizio che non fornisce CF. In questo caso APP IO forzerà il logout dell'utente a valle della risposta ricevuta da BPD.

Sotto processo di allineamento tra PM e TKM

Descrizione processo

Per tutti i wallet cancellati sul PM (sia in caso di cancellazione da App IO che da Issuer), quest'ultimo sarà responsabile di comunicare al sistema TokenManager, il set di strumenti di pagamento per i quali non sarà più necessario recuperare i parametri correlati alla gestione dei token (PAR, tokenPAN). Si precisa che la coda kafka sarà la stessa di quella utilizzata per comunicare a RTD le carte da disattivare.

TK6. Processo di recupero ed associazione delle carte padre con il parametro PAR (HPAN-PAR)

In modalità asincrona, per tutti i CF abilitati, TKM per tutte le carte associate a quel CF invocherà i servizi esposti dai circuiti per recuperare il PAR.

All'atto della creazione del wallet il PAR potrebbe non essere sempre presente, in quanto alcuni circuiti generano tale valore solo in presenza di tokenizzazione del PAN.

Si rende pertanto necessario un meccanismo batch di recupero (schedulato con frequenza da definire) attraverso il quale TKM verificherà, per i wallet sprovvisti di PAR, se dai circuiti è possibile recuperare questa informazione, come conseguenza della generazione di token PAN collegati alla carta padre. Pertanto:

- il modulo TKM per tutte le carte sprovviste di PAR ed afferenti ad utenti che hanno fornito il consenso, invoca i circuiti (Visa, Mastercard, Amex) ed a fronte di PAN, recupera PAR associato;
- successivamente il TKM utilizzando la coda (messaggi su coda) aggiornerà con i nuovi token BPD e/o FA rispetto alla posizione della carta padre/CF. Da questo momento anche HtokenPAN sarà inviato nei flussi verso gli acquirer insieme con HPan.
- BPD riceverà l'informazione sulle azioni da effettuare in merito alle assocazioni PAR e HToken e procederà con l'aggiornamento del DB.

Si precisa che i token non verranno salvati da BPD su CentroStella nel momento in cui verrà inviata una richiesta di enrollment/aggiornamento /disattivazione carta con Token associati. Le richieste verranno indirizzate verso il PM (per procedere con hashing e l'allineamento del TKM), ma non verrà salvata nessuna informazione sui token. I token verranno salvati solo a seguito della ricezione delle associazioni token-PAR dal TKM in quanto sarà quest'ultimo owner del consenso del cittadino all'utilizzo token.

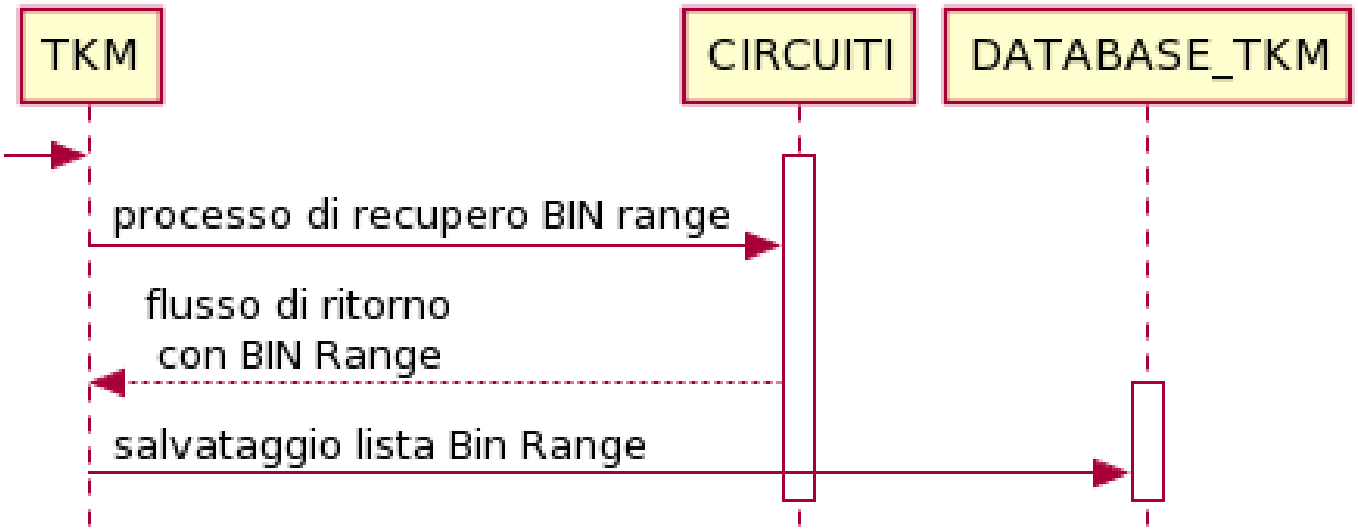
In merito al processo di aggiornamento DB e le varie possibili casistiche si reinvia al TK.9.

TK7. Processi di recupero dai circuiti dei Bin Range

Ogni giorno TKM invoca i servizi esposti dai circuiti per recuperare/aggiornare la lista dei BIN abilitati alla tokenizzazione e salva le informazioni sul proprio DB sulla tabella *BIN:*

- circuito
- BIN range (num)
- dedicato a token (booleano)
- BIN issuer (string)
- Issuer (string)
- timestamp inserimento
- user inserimento
- timestamp update
- user update

Sequence Diagram



TK8. Integrazione Batch Acquirer

Overview generale

Al fine di poter alimentare correttamente il flusso delle transazioni (nella quali verranno aggiunti i parametri HTokenPAN e PAR) il il Batch Acquirer dovrà integrare i seguenti flussi in input:

- la **lista degli strumenti di pagamento enrollati al servizio**, con integrazione degli HTokenPAN.

- la **lista dei PAR enrollati su BPD**: campo attraverso il quale sarà possibile riconoscere ed associare i token figli ad HPAN già attivi su BPD.

Sempre in ambito Batch Acquirer, al fine dipoter armonizzare il processo di enrollmet delle carte tokenizzate, ad oggi alimentato solo dagli Issuer, si renderà necessario aggiungere una terza lista in ingresso ovvero la **lista dei BIN Range**, prodotta dal TKM. La stessa avrà come output un flusso informativo contenente i TokenPAN il quale dovrà essere generato ed inviato dagli Acquirer. In particolare tramite un determinato BIN range (indicato da due PAN che determinano l'inizio e la fine del range), che identifica un gruppo di carte, sarà possibile verificare quali TokenPAN, inviati dagli Acquirer tramite un flusso dedicato (oggetto dello UC TK9"*Inserimento tokenPan sul TKM e recupero PAR"*), appartengono a strumenti già enrollati su BPD. Obiettivo finale sarà quello di creare un processo virtuoso tramite il quale aggiornare la base dati di BPD ed integrare i flussi del Batch Acquirer (ovvero la lista delle carte enrollate).

In aggiunta alla lista di Bin Range, verrà inoltre resa disponibile da parte del TKM una lista di tokenPAN a cui è applicato hashing SHA256 (utilizzando il salt già impiegato per l'hashing dei PAN in CentroStella), che corrispondono a tutti i token per cui è già nota l'associazione tramite PAR. Questo permetterà di ridurre il volume dei file prodotti dal Batch Acquirer, che scarterà tutti tokenPAN già noti, dalla lista da inviare al TKM.

1. Liste HPAN/HTokenPAN e PAR

Produzione File

Entrambe le liste saranno messe a disposizione tramite elaborazione schedulata, ed opzionalmente tramite invocazione di un API per avvio manuale, e saranno prodotte sulla base degli strumenti che sono stati registrati attivamente nella tabella degli strumenti di pagamento attivi in RTD, controllando la tabella *rtd_database.rtd_payment_instrument_data*, che contiene:

- HPAN/TokenPAN
- Flag per abilitazione in BPD
- Flag per abilitazione in FA
- timestamp inserimento
- User inserimento
- timestamp update
- User update
- PAR

Il flusso prevede di aggiornare la tabella tramite il recupero nei database di BPD/FA degli strumenti di pagamento da inserire/aggiornare, sulla base di un delta, definito nella tabella *rtd_database.rtd_exec_date.* Nell'estrazione in delta sono inclusi:

- Attivazione strumento
- Disattivazione strumento
- Aggiornamento PAR

Il processo già esistente viene rivisto per aggiungere il PAR agli strumenti da inserire, oppure da aggiornare per inclusione del PAR mancante in tabella. La struttura della lista di PAR è equivalente a quella della lista HPAN.

Una volta concluso l'aggiornamento della tabella con le informazioni recuperate dai database di BPD/FA viene eseguita una lettura a blocchi della lista di strumenti attivi, da cui verranno prodotti:

- File contenente HPAN/HTokenPAN
- File contenente PAR

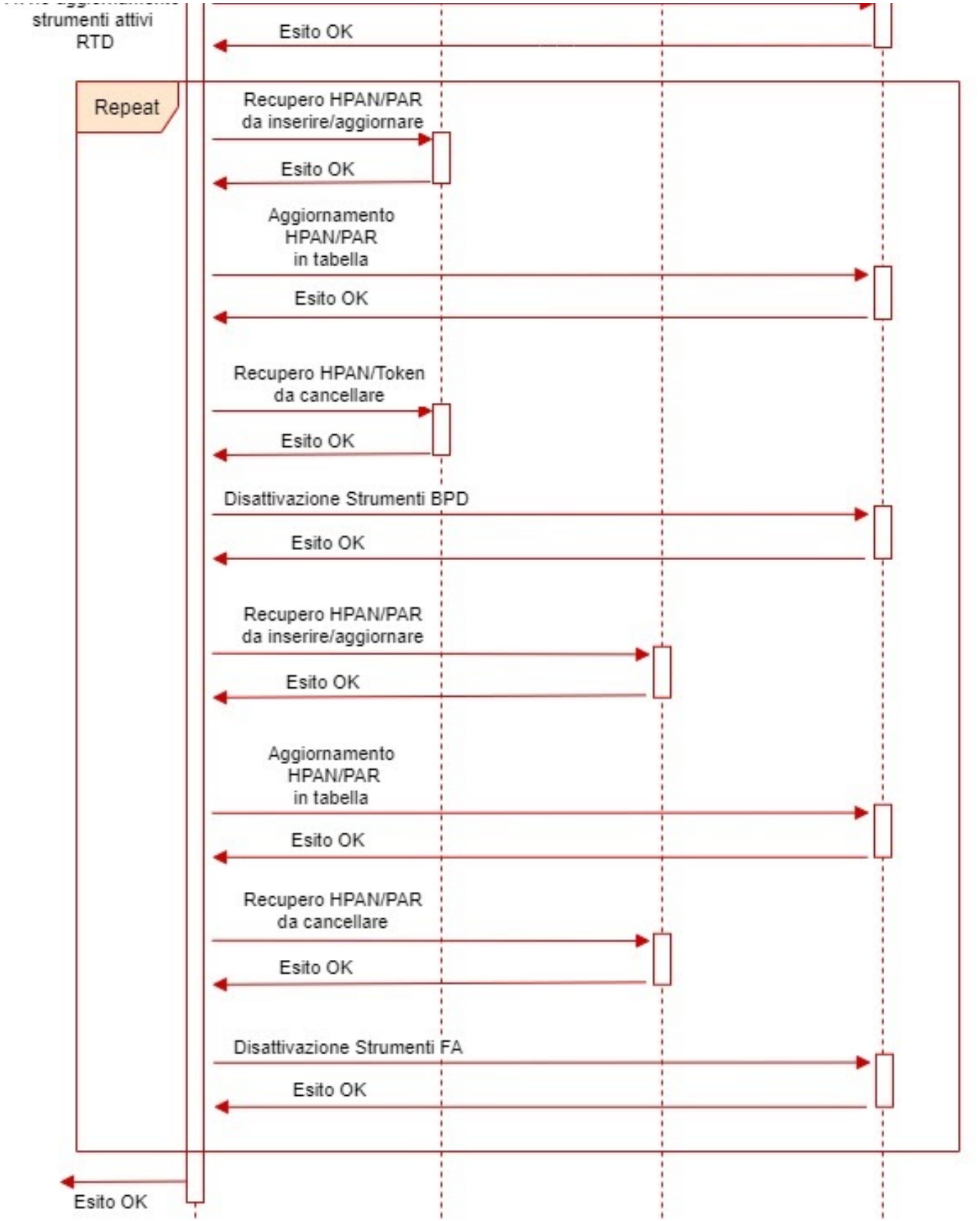
Una volta conclusa la lettura delle informazioni dal database di RTD, per entrambi i file saranno eseguiti i passaggi:

- Compressione del file
- Calcolo della checksum del file compresso, con algoritmo SHA256
- Upload del file su Blob Storage, aggiungendo la checksum calcolata come informazione aggiuntivi

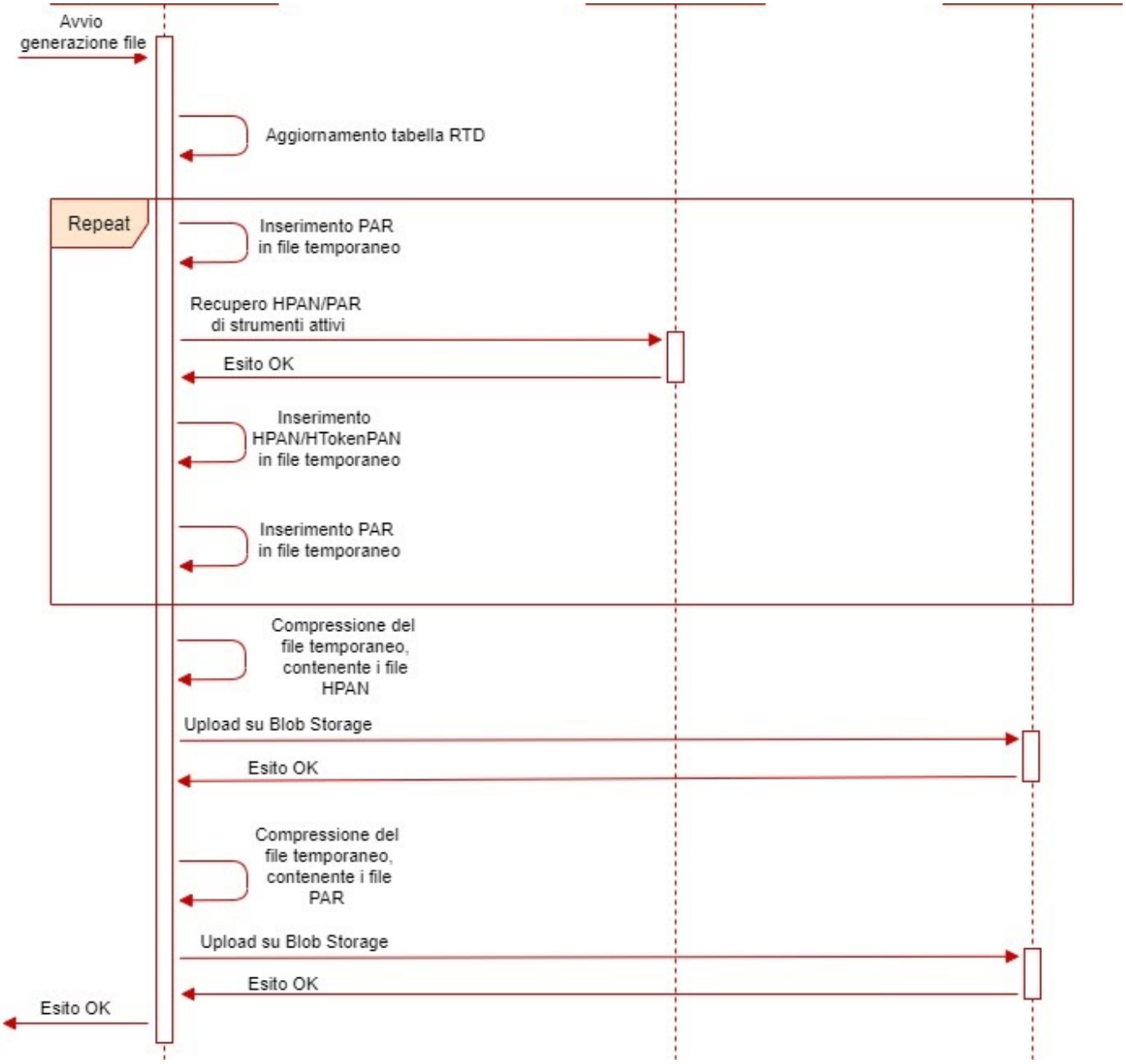
Al termine del processo i file temporanei, impiegati durante l'elaborazione, saranno rimossi.

Sequence Diagrams

Aggiornamento dati su RTD



Generazione file ed upload su storage Azure



Esposizione File

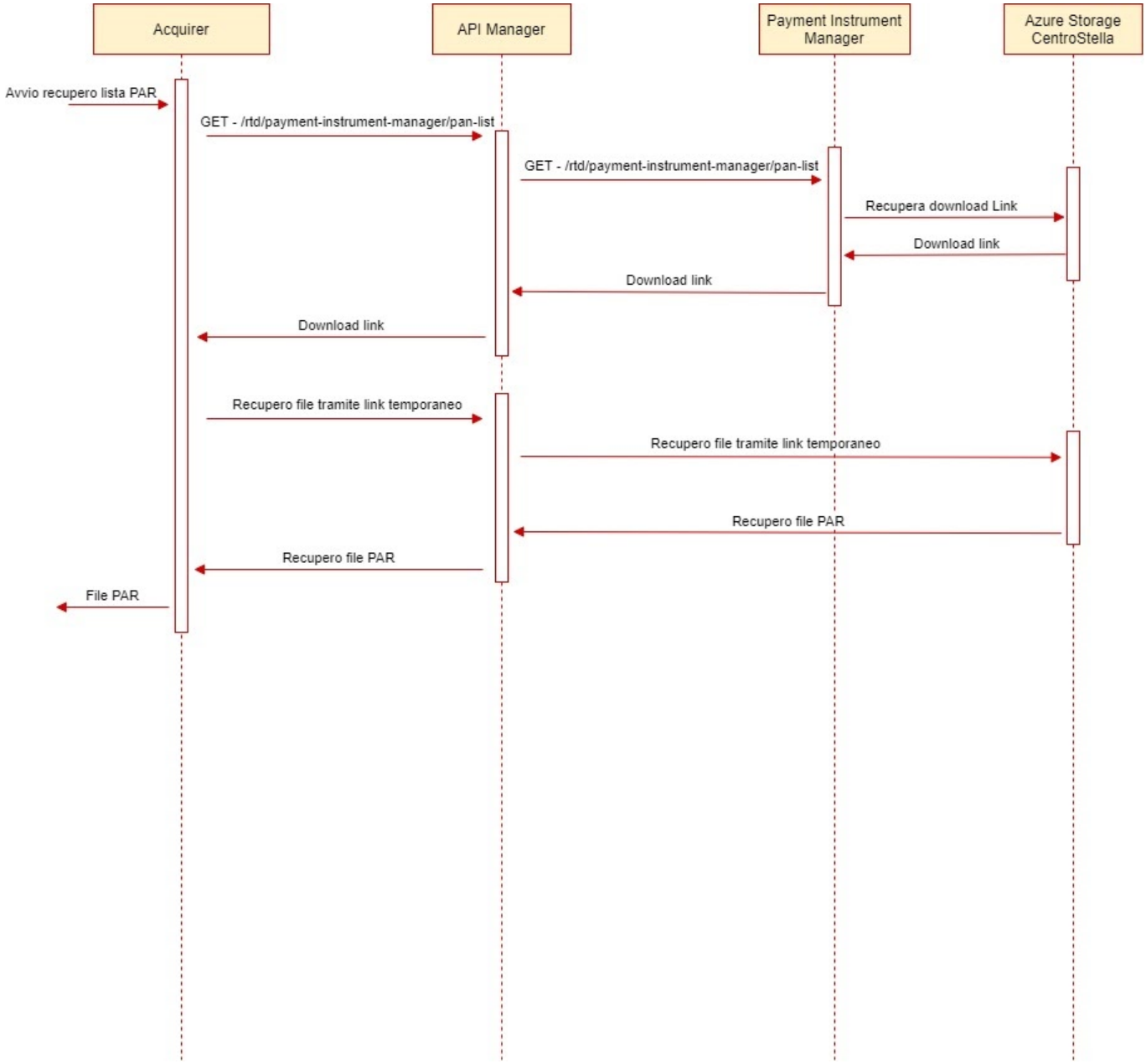
Il file viene reso disponibile per il recupero dal Blob Storage di CentroStella, utilizzando una API nel componente di RTD *Payment Instrument Manager,* che si occuperà di contattare l'Azure Blob Storage, da cui verranno prodotti dei link one-shot, validi per un periodo limitato, che saranno inclusi nella risposta, strutturata come segue:

- fileLinks lista dei link one-shot da cui recuperare i file contenenti i bin-range
- numberOfFiles Indicazione del numero di file
- availableUntil Data indicante la durata massima dei link presenti nella risposta
- fileGenerationFile Data indicante la generazione dei file ottenibili dai link presenti nella risposta

Una volta ottenuta lista dei link da scaricare, nel caso il numero dei file sia diverso da zero e la data di generazione del file sia ancora valida, verrà effettuato il download tramite chiamate sequenziali su tutti i link ottenuti dalla prima chiamata. In caso di errori in ognuna delle fasi di recupero il processo batch viene bloccato.

Sia l'API di recupero della lista, ed i link per il download generati passeranno per l'API Manager di Azure, con la necessità che le chiamate avvengano fornendo una chiave di sottoscrizione valida per il prodotto, ed impiegando un certificato SSL valido, fornito tramite firma da parte di PagoPA, per l'autenticazione su TKM. Sia la chiamata di recupero del link, che quella effettuata per il download effettivo, passeranno attraverso l' API Manager di Azure, con i meccanismi di autenticazione ed autorizzazione già menzionati.

Sequence Diagram



2. Lista dei BIN Range

Produzione File

La lista dei Bin Range sarà prodotta sulla base delle informazioni disponibili al TKM, ed una volta prodotta sarà salvata sull'Azure Blob Storage di riferimento, da cui sarà utilizzata per il recupero da parte degli acquirer. La lista potrà essere suddivisa su più file, nel caso la dimensione massima configurata sia superata. Il formato della lista sarà quella di un file csv contenente i seguenti due campi, delimitati da ";" :

- rangeStart numerico, PAN da cui inizia il range per un circuito
- rangeEnd numerico, PAN da cui termina il range per un circuito

Assieme ai file saranno salvate le informazioni contenenti la data di creazione dei file, da utilizzare eventualmente nel batch per validazione, assieme alla checksum in SHA-256 della versione compressa del file contenente la lista.

Api Rest - Recupero lista file Bin Range

Path: /tkm/acquirerm/binrange/link (Nota: in UAT il base URI è /tkm/uat/acquirerm….)

Method: GET

Path Parameters

Nessun parametro

Query Parameters

Nessun parametro

Request Header

Campo	Tipo	Obbligatorio	Descrizione
Ocp-Apim-Subscription-Key	Alfanumerico	SI	Chiave di sottoscrizione da utilizzare per il servizio

Request Body

Nessun campo

Response Code: 200 (OK)

Response Body

Campo	Formato	Obbligatorio	Descrizione
fileLinks	List<Alfanumerico>	SI	Lista di link temporanei da cui recuperare i file
numberOfFiles	Numerico	SI	Numero di file da scaricare
availableUntil	Timestamp	SI	Timestamp che indica la data di scadenza per la validità dei link
generationDate	Timestamp	SI	Timestamp che indica la data in cui i file da scaricare sono stati aggiornati. Utilizzata per validare i file da scaricare nel processo batch

Response Header

Campo	Formato	Obbligatorio	Descrizione
Request-Id	Alfanumerico	NO	Request ID, identificativo univoco del sistema (UUID)

HTTP Error Codes

Lista di messaggi d'errore associati al servizio

HTTP Response Code	Error code	Description
404	FILE_NOT_FOUND	file not found
500	GENERIC_ERROR	generic error
403	AUTHENTICATION_ERROR	Authentication error
401	AUTHORIZATION_ERROR	Authorization error

Esposizione File

Il file viene reso disponibile per il recupero dall'Azure Blob Storage, utilizzando una API associata al TKM, che si occuperà di contattare l'Azure Blob Storage, da cui verranno prodotti dei link one-shot, validi per un periodo limitato, che saranno inclusi nella risposta, strutturata come segue:

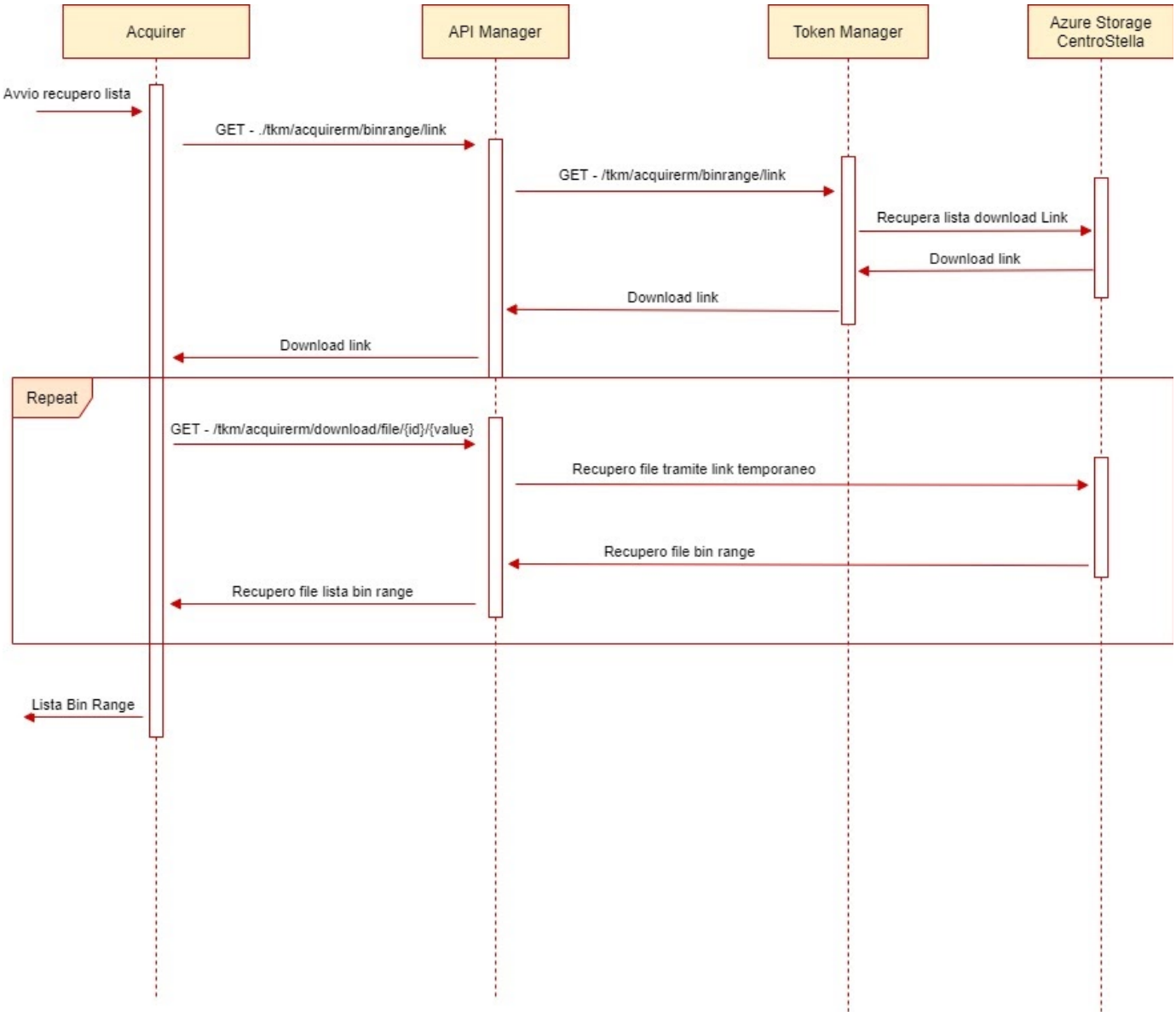
- fileLinks lista dei link one-shot da cui recuperare i file contenenti i bin-range
- numberOfFiles Indicazione del numero di file
- availableUntil Data indicante la durata massima dei link presenti nella risposta

- fileGenerationFile Data indicante la generazione dei file ottenibili dai link presenti nella risposta

Una volta ottenuta lista dei link da scaricare, nel caso il numero dei file sia diverso da zero e la data di generazione del file sia ancora valida, verrà effettuato il download tramite chiamate sequenziali su tutti i link ottenuti dalla prima chiamata. In caso di errori in ognuna delle fasi di recupero il processo batch viene bloccato.

Sia l'API di recupero della lista, ed i link per il download generati passeranno per l'API Manager di Azure, con la necessità che le chiamate avvengano fornendo una chiave di sottoscrizione valida per il prodotto, ed impiegando un certificato SSL valido, fornito tramite firma da parte di PagoPA, per l'autenticazione su TKM. Sia la chiamata di recupero del link, che quella effettuata per il download effettivo, passeranno attraverso l' API Manager di Azure, con i meccanismi di autenticazione ed autorizzazione già menzionati.

Sequence Diagram



3. Lista HTokenPAN

Produzione File

La lista contenente gli hash dei tokenPAN sarà prodotta sulla base delle informazioni disponibili al TKM, recuperando tutti i tokenPAN per cui è già nota l'associazione tramite PAR ad una carta. Una volta prodotta sarà salvata sull'Azure Blob Storage di riferimento, da cui sarà utilizzata per il recupero da parte degli acquirer. Il formato di un file sarà quella di un file csv contenente un singolo campo per record, valorizzato con l'hash di un tokenPAN per cui è nota l'associazione al TKM. La lista potrà essere suddivisa su più file, nel caso la dimensione massima configurata sia superata. Assieme ai file saranno salvate le informazioni contenenti la data di creazione dei file, da utilizzare eventualmente nel batch per validazione, assieme alla checksum in SHA-256 della versione compressa del file contenente la lista.

Api Rest - Recupero lista file HTokenPAN

Path: /tkm/acquirerm/htoken/known/link (Nota: in UAT il base URI è /tkm/uat/acquirerm….)

Method: GET

Path Parameters

Nessun parametro

Query Parameters

Nessun parametro

Request Header

Campo	Tipo	Obbligatorio	Descrizione
Ocp-Apim-Subscription-Key	Alfanumerico	SI	Chiave di sottoscrizione da utilizzare per il servizio

Request Body

Nessun campo

Response Code: 200 (OK)

Response Body

Campo	Formato	Obbligatorio	Descrizione
fileLinks	List<Alfanumerico>	SI	Lista di link temporanei da cui recuperare i file
numberOfFiles	Numerico	SI	Numero di file da scaricare
availableUntil	Timestamp	SI	Timestamp che indica la data di scadenza per la validità dei link
generationDate	Timestamp	SI	Timestamp che indica la data in cui i file da scaricare sono stati aggiornati. Utilizzata per validare i file da scaricare nel processo batch

Response Header

Campo	Formato	Obbligatorio	Descrizione
Request-Id	Alfanumerico	NO	Request ID, identificativo univoco del sistema (UUID)

HTTP Error Codes

Lista di messaggi d'errore associati al servizio

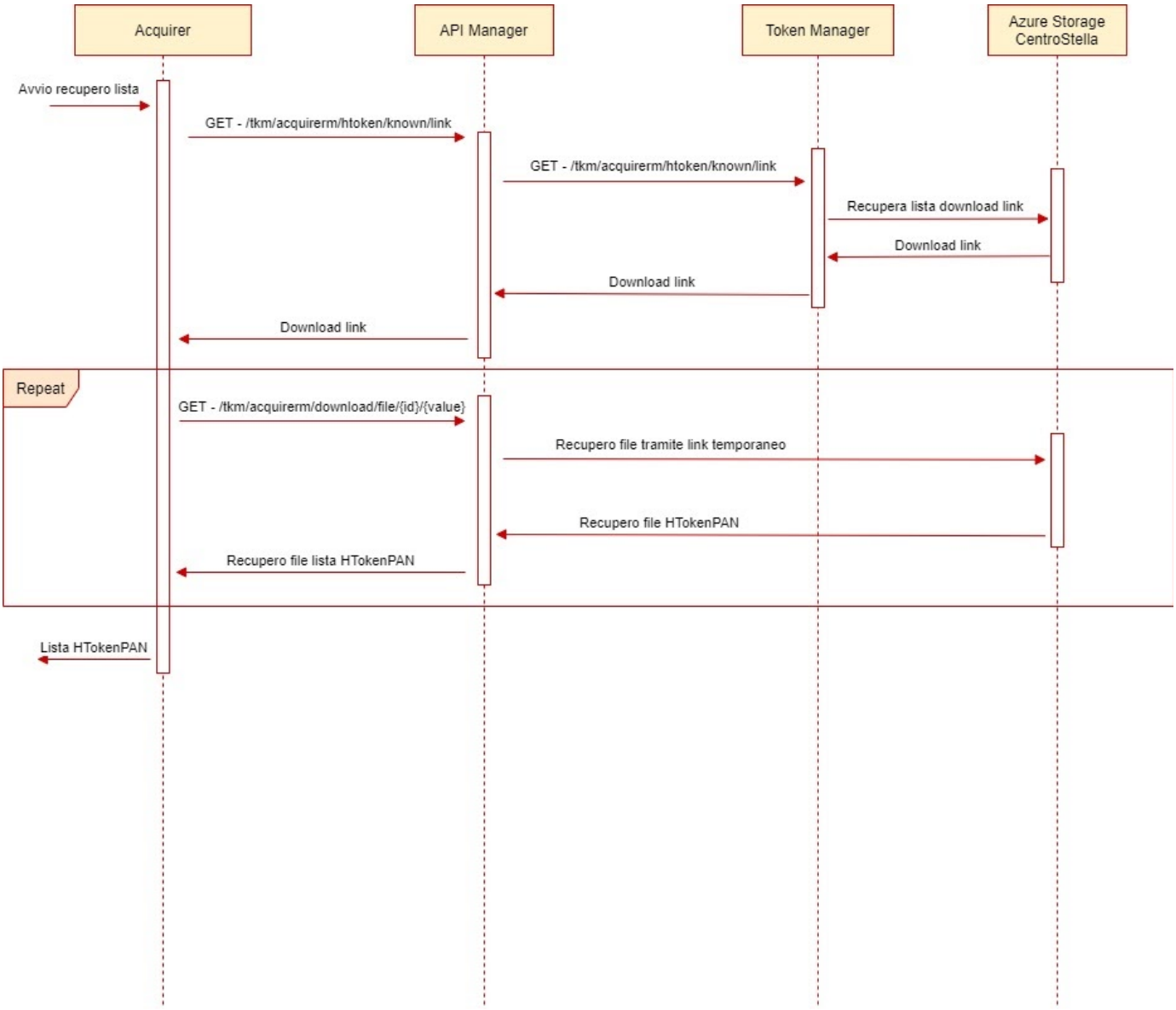
HTTP Response Code	Error code	Description
404	FILE_NOT_FOUND	file not found
500	GENERIC_ERROR	generic error
403	AUTHENTICATION_ERROR	Authentication error
401	AUTHORIZATION_ERROR	Authorization error

Esposizione File

Il file viene reso disponibile per il recupero dall'Azure Blob Storage, utilizzando una API associata al TKM, che si occuperà di contattare l'Azure Blob Storage, da cui verrà prodotto un link one-shot, valido per un periodo limitato, che sarà comunicato nella risposta del servizio, e da cui sarà possibile ottenere il file.

Sia l'API ed il link di recupero generato passeranno per l'API Manager di Azure, con la necessità che le chiamate avvengano fornendo una chiave di sottoscrizione valida per il prodotto, ed impiegando un certificato SSL valido, fornito tramite firma da parte di PagoPA, per l' autenticazione su TKM. Sia la chiamata di recupero del link, che quella effettuata per il download effettivo, passeranno attraverso l'Api Manager di Azure, con i meccanismi di autenticazione ed autorizzazione già menzionati.

Sequence Diagram



Path: /tkm/acquirerm/download/file/{id}/{value} (Note: in UAT the base URI is /tkm/uat/acquirerm….)

Method: GET

Path Parameters

Campo	Formato	Obbligatorio	Descrizione
id	Alfanumerico	NO	Id identificante il tipo del file temporaneo
value	Alfanumerico	NO	Valore identificante la porzione di file

Query Parameters

Nessun parametro

Request Header

Campo	Formato	Obbligatorio	Descrizione
Ocp-Apim-Subscription-Key	Alfanumerico	YES	Chiave di sottoscrizione per il servizio

Response Body

File compresso contenente il .csv file richiesto. A seconda del metodo inizialmente invocato per ottenere la lista il file conterrà la lista di Bin Ranges, o la lista dell' HTokenPANs

Response Code: 200 (OK).

Response Header

Campo	Formato	Obbligatorio	Descrizione
Request-Id	Alfanumerico	NO	Request ID, identificativo univoco del sistema (UUID)
Checksum-Sha256	Alfanumerico	NO	SHA-256 checksum del file compresso. Utilizzato nel batch per validazione

HTTP Error Codes

HTTP Response Code	Error code	Description
404	FILE_NOT_FOUND	file not found
500	GENERIC_ERROR	generic error
403	AUTHENTICATION_ERROR	Authentication error
401	AUTHORIZATION_ERROR	Authorization error

TK9. Inserimento TokenPAN sul TKM e recupero PAR

Tutti i token ricevuti all'interno del flussi TokenPAN ed elaborati ed inviati dagli Acquirer, se compresi all'interno della lista di BIN range, vengono salvati sulla tabella *TokenCard* del TKM. in modo asincrono TKM invocherà i circuiti (Visa, Mastercard,Amex) al fine di recuperare per ciascun TokenPAN, il PAR associato.

Il modulo TKM salverà il parco informativo sul proprio DB sulla tabella *TokenCard*:

- tokenPAN (cifrato) - PRIMARY KEY
- ultime 4 cifre del PAN
- HTokenPAN (hash del valore con salt richiesto tramite API a PM)
- PAR
- BIN issuer (string)
- timestamp inserimento
- user inserimento
- timestamp update
- user update

Un volta recuperato il PAR associato al Token, TKM verifica la presenza di record nella tabella *Cards* con il medesimo PAR.

In caso positivo il TKM inverà le informazioni, utilizzando una coda dedicata (La coda sarà la stessa impiegata per l'UC TK6), tramite cui sarà effettuato l'aggiornamento dei nuovi token BPD e/o FA rispetto alla posizione della carta padre/CF. Da questo momento anche HtokenPAN sarà inviato nei flussi verso gli acquirer insieme con HPan. Il TKM restituirà al Centro Stella Htoken.

Il contenuto del messaggio inviato a Centro Stella è:

- taxCode codice fiscale della carta padre. ***il campo è obbligatorio***
- timestamp data indicata per l'evento, utilizzata per discriminare eventuali indicazioni fuori ordine. **il campo è obbligatorio**
- cards lista di carte da aggiornare, la struttura dei dati di carta è la seguente:
    - par par della carta padre - **il campo è obbligatorio**
    - action enum contente i valori fissi INSERT_UPDATE o REVOKE. Il valore indica il tipo di aggiornamento puntuale da eseguire sul PAR - **il campo è obbligatorio**
    - hpan hpan della carta padre - **il campo è obbligatorio**
    - htokens token pan da associare alla carta padre
    - haction enum contente i valori fissi INSERT_UPDATE o DELETE. Il valore indica il tipo di aggiornamento puntuale da eseguire sul token

Il processo in funzione all'interno di CentroStella per gestire gli eventi inviati dal TKM, contenenti i tracciati strutturati come sopra, sarà equivalente per gli UC del TK6 e TK9.

Il contenuto della lista cards corrisponde alle carte da aggiornare, associate ad uno specifico codice fiscale, indicato nel campo *taxCode.* Per ogni carta all'interno della lista viene gestito il seguente processo:

- Viene estratta la carta corrispondente al campo obbligatorio *hpan.* Se non presente viene sollevata eccezione per la carta indicata, e l' elaborazione per questa particolare carta è interrotto, procedendo con le successive.
- Viene verificato che il codice fiscale della carta estratta sia corrispondente a quello presente nel campo *taxCode,* altrimenti viene bloccata l'elaborazione per la carta indicata, procedendo con le successive.
- Viene verificato che il campo PAR della carta sia valorizzato nel database BPD:
    - Se il PAR è presente, ma risulta diverso dal valore presente nel tracciato Verrà generato un errore, e verrà interrotta l' elaborazione proseguendo con le successive.
    - Se il PAR è presente e risulta uguale al valore indicato nel tracciato (Punto relativo all'UC TK6) verranno effettuate le operazioni di aggiornamento descritte sotto:

	messaggio TKM				***CentroStella***
ID	***campo PAR***	***STATO Par (Boolean)***	***campo Htokens***	***STATO Htokens***	***Processo di aggiornamento a DB***
1	presente	INSERT_UPDATE	presente	INSERT_UPDATE	• Inserire nuovo record per il PAR impostando lo stato "Active" se il PAR non risulta presente a DB. • Se il PAR è presente a DB ed è in stato "Inactive", si aggiorna lo stato in "Active". • Se il PAR è presente a DB, non sono aggiornati i campi relativi al PAR, e viene riportato l' aggiornamento della data di ultimo evento ricevuto solo se maggiore di quella presenta DB. • I token verranno aggiunti come nuovi record con stato "Active" qualora non dovessero risultare presenti a DB. • I Token presenti a DB in stato "Inactive" saranno impostati in "Active", se la data di ultimo aggiornamento sarà inferiore a quella riportata nel campo *timestamp*
2	presente	INSERT_UPDATE	presente	DELETE	• Inserire nuovo record per il PAR impostando lo stato "Active" se il PAR non risulta presente a DB. • Se il PAR è presente a DB ed è in stato "Inactive", si aggiorna lo stato in "Active". • Se il PAR è presente a DB, non sono aggiornati i campi relativi al PAR, e viene riportato l' aggiornamento della data di ultimo evento ricevuto solo se maggiore di quella presenta DB. • Disattivare i token inviati nel messaggio, se attivi (impostando stato "Inactive"), solo se la data di ultimo aggiornamento sarà inferiore a quella riportata nel campo *timestamp* • Aggiungere record per i Token ricevuti non presenti a DB, impostando lo stato in "Inactive"
3	presente	INSERT_UPDATE	non presente	-	• Aggiungere esclusivamente un record per il PAR ricevuto (se non presente a DB) • Se il PAR risulta presente a DB con stato "Inactive", quest' ultimo verrà aggiornato in "Active"

					• Se il PAR è presente a DB, non sono aggiornati i campi relativi al PAR, e viene riportato l' aggiornamento della data di ultimo evento ricevuto solo se maggiore di quella presenta DB.
4	presente	REVOKE	presente/non presente	DELETE oppure *blank*	• Se l'ultima data di aggiornamento tramite evento del PAR è inferiore a quella indicata nel campo timestamp Disattivazione PAR e a cascata di tutti i token ad esso associati (a prescidere dai token contenuti eventualmente nella lista tokens)

Gestione errori

Nel caso di errori diversi dai casi applicativi gestiti internamente al processo (crash connessione a db, errori temporanei di connettività), l'evento verrà bloccato non dando l'acknowledgment dell'avvenuta elaborazione, e tutte le modifiche sulle entità fino a quel momento gestite saranno mandate in roll-back. L'evento sarà riprocessato dal componente di CentroStella dedicato alla gestione degli strumenti di pagamento.

TK10. Integrazione Acquirer

L'acquirer dovrà inviare al Centro Stella i seguenti tracciati, prodotti dal processo batch. I due tracciati sono entrambi prodotti a partire dal flusso delle transazioni, già impiegato come input nella procedura batch già in produzione. **I due tracciati potranno essere prodotti anche utilizzando il flusso delle transazioni privo del nuovo campo**:

1. flusso delle transazioni as-is con l'aggiunta del campo PAR all'interno del flusso
2. file cifrato pgp con la lista dei TokenPAN (non hashato) associato al BIN range inviato tramite il Batch

Descrizione del processo

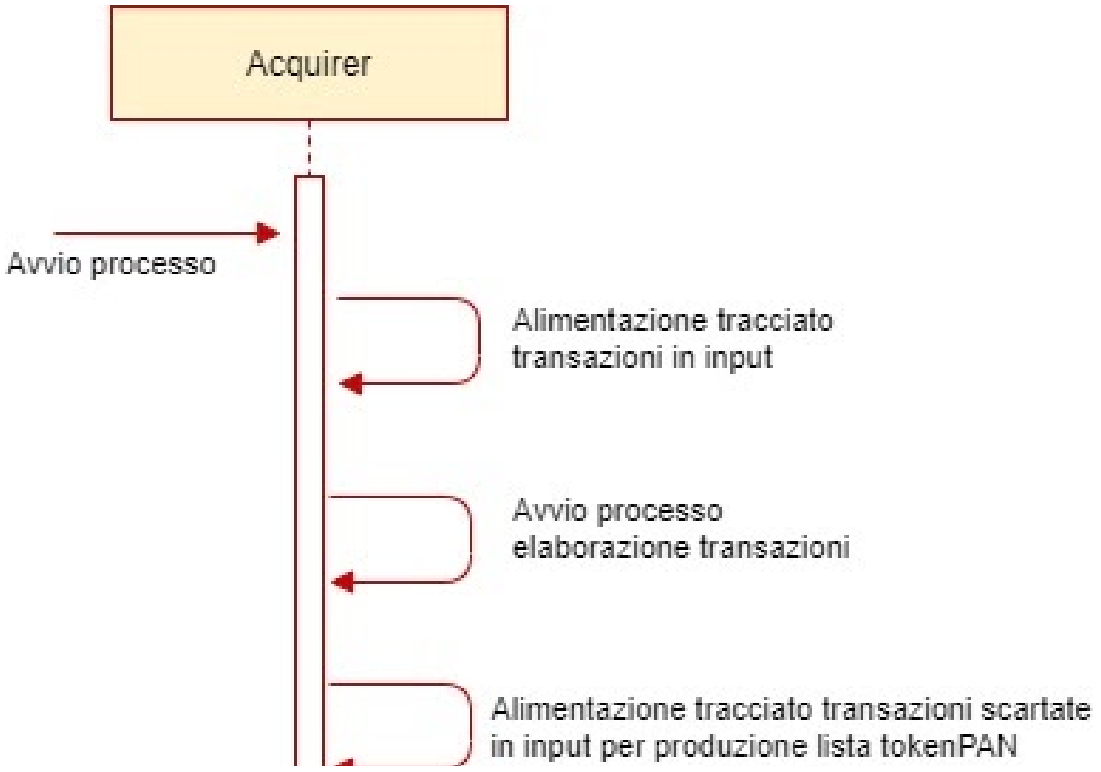
Overview Generale

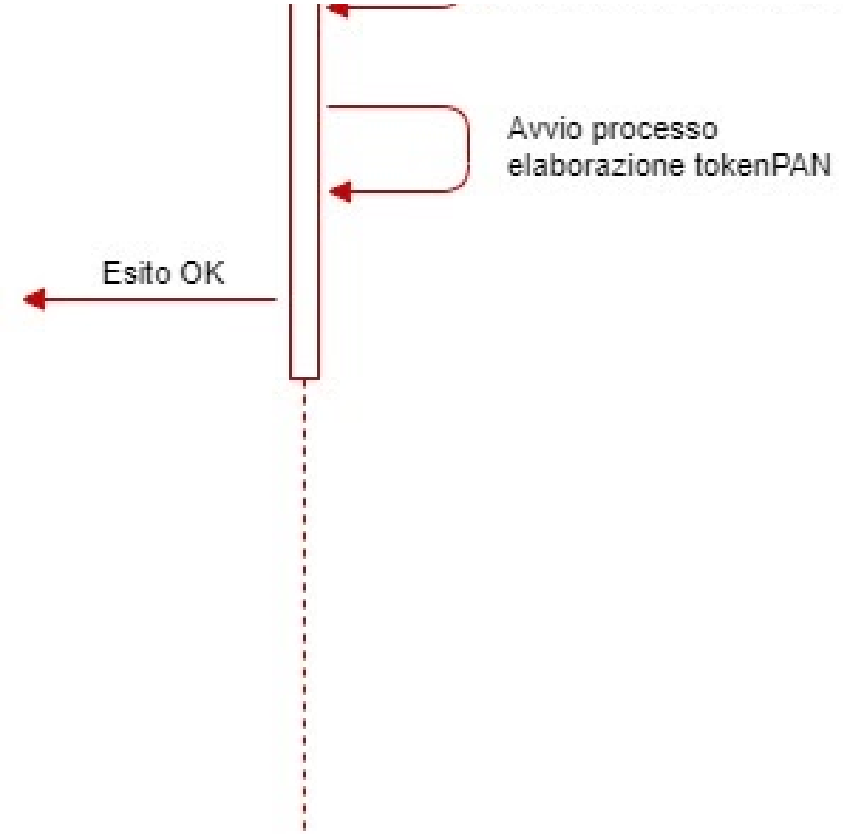
I due file da inviare saranno prodotti a partire dal tracciato giornaliero delle transazioni gestite dagli acquirer. La produzione dei due file viene gestita tramite due fasi nello stesso processo, in cui saranno prodotti i due tracciati da inviare a CentroStella/TKM.

Le due elaborazioni avverranno in sequenza, e la produzione del file di output relativo alle transazioni si concluderà prima dell'avvio della produzione del file dei tokenPAN, che sarà prodotta sulla base degli scarti nella prima fase, avvenuti per mancanza di HPAN/PAR.

Nel caso il processo s'interrompa nel corso dell'elaborazione del tracciato dei tokenPAN l'elaborazione delle transazioni NON sarà intaccata, ed il tracciato da confrontare potrà essere riutilizzato indipendentemente dalla produzione di nuovi tracciati di output per le transazioni da inviare a CentroStella.

Sequence Diagram





Parte 1 - Acquisizione Transazioni Acquirer

L'elaborazione delle transazioni avviene secondo il processo già definito, rivisto per considerare l'aggiunta nel tracciato di un nuovo campo non obbligatorio, contenente il PAR, che sarà utilizzato in aggiunta al controllo degli HPAN per definire quali transazioni nei flussi degli acquirer saranno inviate a CentroStella.

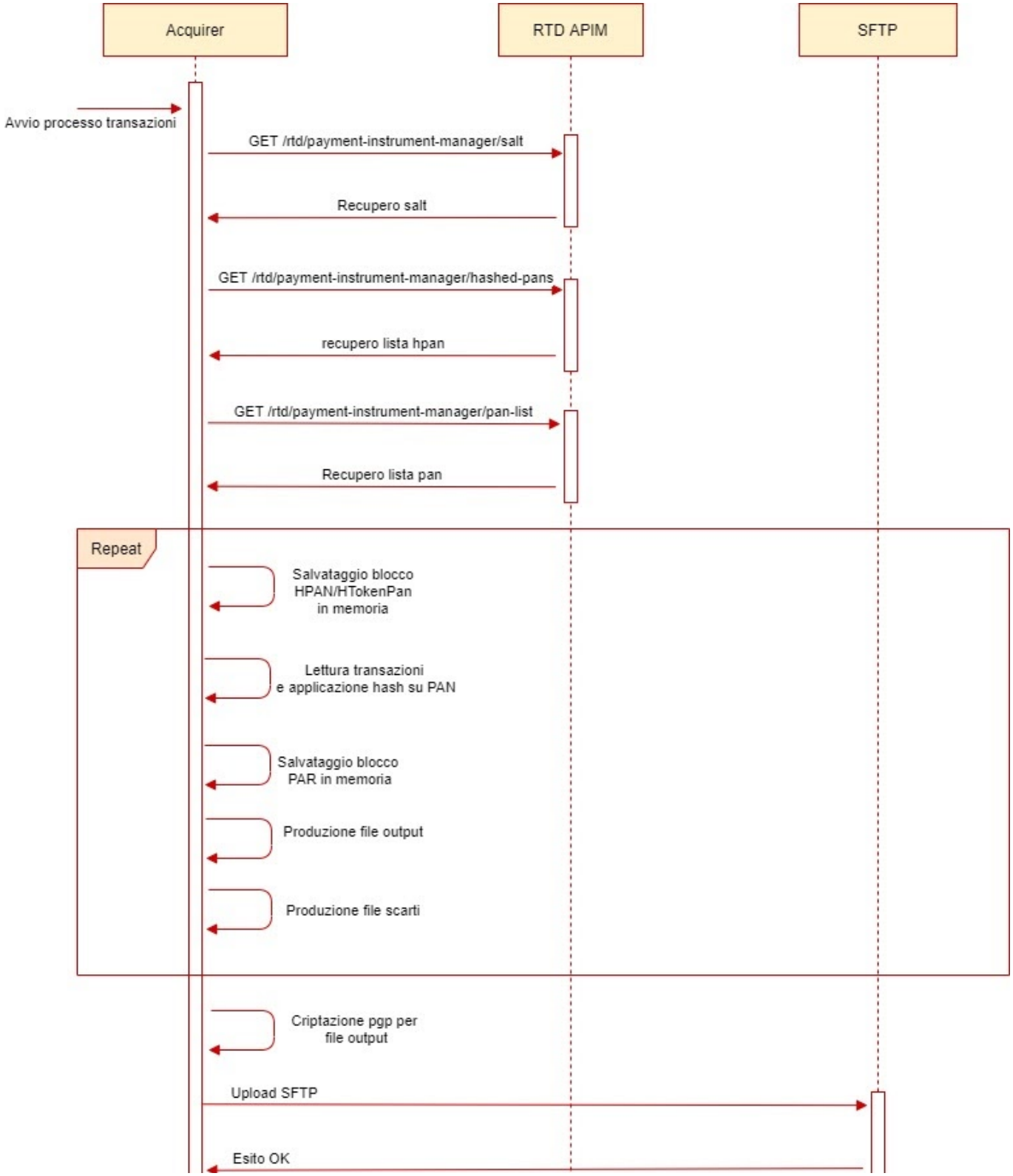
L'adattamento da parte di tutti i soggetti acquirer al processo di elaborazione del nuovo tracciato non sarà immediatamente richiesta nel caso non siano trattate transazioni per cui è utilizzato un PAR, e potrà essere utilizzato il processo già in funzione, inviando il formato del tracciato privo del nuovo campo, che continuerà ad essere riconosciuto come valido all'interno dei processi di acquisizione dei flussi in CentroStella.

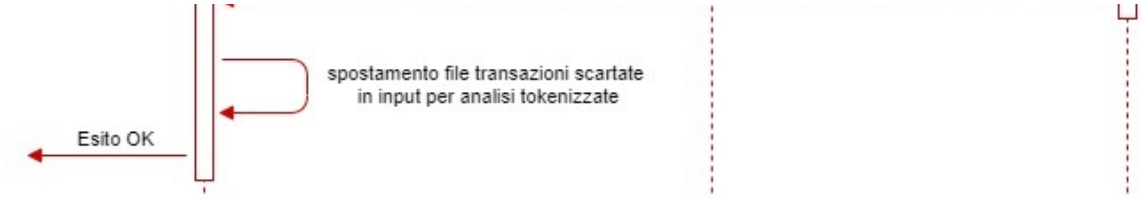
Il processo di invio delle transazioni verso la Piattaforma CentroStella (RTD) è rivisto nelle seguenti fasi:

- La Piattaforma CentroStella genera un flusso contenente gli HPAN enrollati al servizio CentroStella (vedere UC TK8)
- La Piattaforma CentroStella genera un flusso contenente i PAR degli strumenti di pagamento/token enrollati al servizio CentroStella (vedere UC TK8)
- L'Acquirer consolida i dati relativi a tutte le transazioni di interesse contabilizzate nell'ultimo ciclo di regolamento verso l'esercente. Provvede pertanto alla generazione di un file di testo in formato csv (con naming file e tracciato dettagliati nel paragrafo dedicato) e lo deposita su una folder su cui è in polling il batch. Il deposito del file è il trigger che fa partire il processo di elaborazione del batch.
- Il Batch invoca il servizio esposto dalla Piattaforma tramite il quale viene generato un link one shot e attivo temporalmente per effettuare il download del flusso HPAN enrollati ai servizi della piattaforma CentroStella.
- Il Batch invoca il servizio esposto dalla Piattaforma tramite il quale viene generato un link one shot e attivo temporalmente per effettuare il download del flusso di PAR degli strumenti enrollati ai servizi della piattaforma CentroStella.
- Il Batch chiama il servizio esposto da CentroStella per ottenere la chiave di hashing costante da aggiungere al PAN per effettuare l' hashing dei PAN contenuti nel flusso delle transazioni.
- Il Batch legge entrambi i flussi in input (lista HPAN e transazioni) e, per ogni riga del file di transazione effettua l'hash del PAN. Il contenuto dei file sarà tenuto temporaneamente in una struttura dati in-memory. Nel caso la dimensione dei file sia di dimensione eccedente il valore massimo, definito a livello di configurazione applicativa, solo una parte del contenuto sarà tenuta in memoria.
- HPAN e PAR (se presente) vengono confrontati rispettivamente con i dati recuperati dalle lista HPAN e PAR. **Una transazione è completamente scartata se non esiste un match all'interno dei flussi di HPAN e PAR**
- Nel caso in cui non siano stati tenuti in memoria tutti i dati dalla lista HPAN e PAR, i punti precedenti saranno ripetuti, caricando in memoria la parte rimanente, ed utilizzando le transazioni scartate nella precedente esecuzione come input. **Il file prodotto sarà il risultato di tutte le elaborazioni effettuate**.
- Le transazioni considerate da scartare al termine dell'elaborazione saranno utilizzate per estrarre TokenPAN e PAR, dove il campo corrispondente per il secondo valore risulti valorizzato. Viene prodotto un file contenente unicamente i campi di associazione tokenPAN, Circuito, e PAR. Il tracciato prodotto che sarà impiegata per la produzione della lista di TokenPAN, confrontata con la lista recuperata dal TKM

- Il Batch, conclusa l’elaborazione al punto precedente, termina la scrittura del flusso filtrato in output
- I file originali per le transazioni sono archiviati
- I file temporanei utilizzati nel corso dell’elaborazioni sono eliminati
- Il Batch effettua la cifratura PGP del flusso di output.
- Il Batch deposita il flusso delle transazioni filtrato su sFTP di CentroStella.
- Il processo di elaborazione delle transazioni termina, e viene avviata l’elaborazione per confronto con la lista di Bin Range

Sequence Diagram





Formato del tracciato

Di seguito vengono descritti i dettagli relativi al Flusso Standard PagoPA.

La naming convention del file è la seguente:

[servizio].[ABI].[tipofile].[data].[ora].[nnn].csv

in particolare:

servizio: fisso a 'CSTAR' (5 digit alfanumerico)

ABI: ABI del mittente (5 digit numerico)

tipofile: fisso a tipologia di file (6 digit alfanumerico)

nnn: progressivo file (3 digit numerico)

Campo	Formato	Note
servizio	Alfanumerico - 5 char	valore fisso CSTAR
ABI	Alfanumerico - 5 char	codice ABI del mittente
tipo_file	Alfanumerico - 6 char	tipologia del flusso inviato. Valore fisso a TRNLOG
[data].[ora]	YYYYMMDD.HHMISS	timestamp di creazione del file
nnn	Alfanumerico - 3 char	Valore progressivo del file (es. 001)

Si precisa che:

Il file è in formato .csv, con separatori “;”

il file è cifrato con chiave pubblica pgp rilasciata da PagoPa SpA

Il contenuto del file non prevede record di testa e coda ma solo record di dettaglio, secondo questo tracciato:

Campi presenti nel Flusso Standard PagoPA.

Campo	Tipo	Obbligatorio	Note
codice_acquirer	Alfanumerico - max 20 char	SI	Codice ABI della banca Acquirer.
tipo_operazione	Alfanumerico - regexp [0-9]{2}	SI	Tipo operazione: 00 - pagamento 01 - storno pagamento 02 - pagamento con ApplePay 03 - pagamento con GooglePay xx - usi futuri La tipologia 02 e 03 non sempre risulterà valorizzata dagli Acquirer
tipo_circuito	Alfanumerico - regexp [0-9]{2}	SI	Payment circuit: 00 – Pagobancomat

			• Le transazioni su questo circuito verranno inviate esclusivamente dall'Acquirer Bancomat. 01- Visa 02- Mastercard 03- Amex • Le transazioni su questo circuito verranno inviate esclusivamente dall'Acquirer AMEX. 04- JCB 05- UnionPay 06- Diners • Le transazioni su questo circuito verranno inviate esclusivamente dall'Acquirer Diners. 07- PostePay Code 08- BancomatPay 09- SatisPay 10- Circuiti Privati(onus, owen) xx - Usi Futuri
hash_pan	Alfanumerico – max 64 char	SI	Hash del PAN dello strumento di pagamento utilizzato. Nel caso di circuito non card based rappresenta l'identificativo univoco dello strumento di pagamento privativo, che l'utente può registrare attraverso App IO o touch point della banca Issuer.
date_time	DateFormat *FORMATO ISO8601 yyyy-MM-ddTHH:mm:ss.SSSXXXXX*	SI	Timestamp dell'operazione di pagamento effettuata presso l' Esercente. Si precisa che non è sempre disponibile il dettaglio in merito al secondo per tutte le transazioni. In tale circostanza, il dettaglio sarà paddato con tutti '0'
id_trx_acquirer	Alfanumerico – max 255 char	SI	Identificativo univoco della transazione a livello di Acquirer. può essere popolato con l'*ARN*, oppure nel caso in cui tale dato non fosse presente, con un *id univoco* che permette di identificare univocamente la transazione lato Acquirer.
id_trx_issuer	Alfanumerico – max 255 char	NO	Codice autorizzativo rilasciato dall' Issuer (es: AuthCode)
correlation_id	Alfanumerico – max 255 char	NO	Identificativo di correlazione fra operazione di pagamento ed eventuale storno/reversal. In certi casi, il dato non può essere recuperato dall' Acquirer e l' informazione non sarà inviato nel campo in questione
total_amount	Numerico	SI	Valorizzato in centesimi di euro (es: 10€ = 1000) ed espresso in valore assoluto: il segno è dedotto dal tipo operazione *"00-pagamento, 01-storno "*
currency	Alfanumerico - max 3 char	NO	Valore fisso 978 = EUR. Si utilizza codifica internazionale ISO.
acquirer_id	Alfanumerico – max 255 char	SI	Identificativo univoco dell'Acquirer. Nel caso di transazione con carta rappresenta il valore omonimo veicolato nei tracciati dei circuiti internazionali. • Nel circuito Pagobancomat corrisponde al campo *codice_sia_abi* • Circuito Visa/Mastercard: *acquirer_id* In altri casi il campo sarà valorizzato con un dato fisso a seconda dell' Acquirer di riferimento
merchant_id		SI	

	Alfanumerico – max 255 char		Identificativo univoco del negozio fisico presso l'Acquirer (noto anche all'Esercente ed utilizzato dallo stesso per registrarsi alla piattaforma di Fatturazione Automatica). Nel circuito Pagobancomat può corrispondere al campo: esercente
terminal_id	Alfanumerico – max 255 char	SI	Identificativo del terminale/POS (Point of Sale) presente presso l' Esercente. • Nel circuito Pagobancomat corrisponde al campo: *stabiliment o cassa* • Circuito Visa/Mastercard: *terminal_id*
bank_identification_number (BIN)	Alfanumerico – regexp [0-9]{6}|[0-9]{8}	SI	Codice contenente le prime 8 cifre dello strumento di pagamento. • Nel circuito Pagobancomat corrisponde al campo: *codice_abi*
MCC	Alfanumerico – max 5 char	SI	Merchant Category Code.
PAR	Alfanumerico	NO	Payment Account Reference Il campo deve contenere l'informazione del PAR, che può essere definita come un collettore capace di associare ogni TokenPAN al PAN della carta fisica, grazie all'associazione unica ed immutabile fra PAN e PAR, e Token e PAR

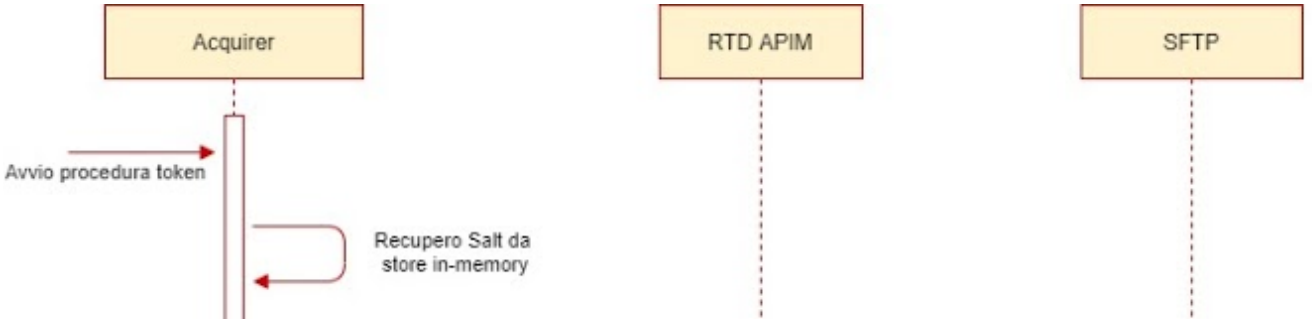
Parte 2 - Acquisizione TokenPAN Acquirer

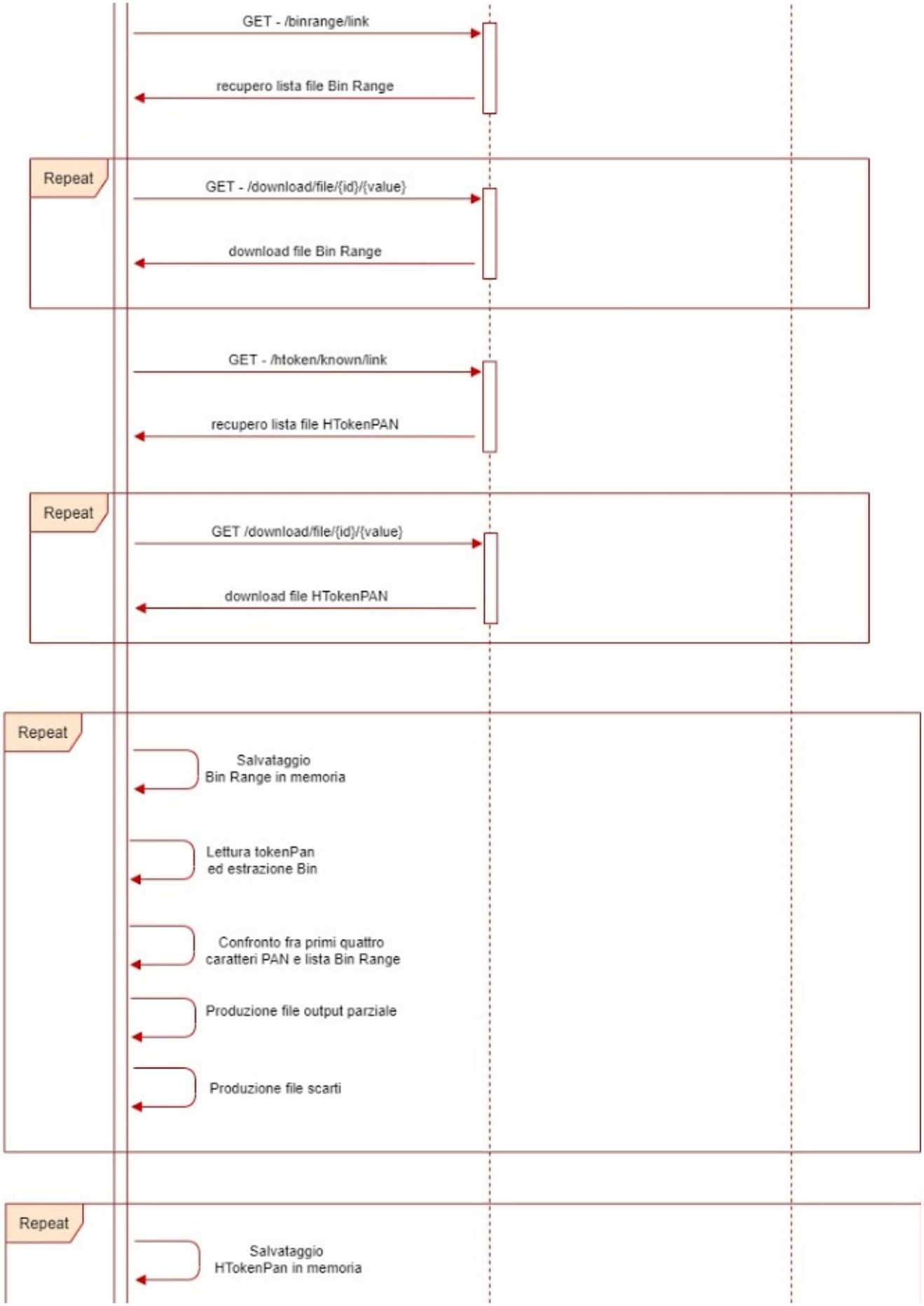
La produzione della lista di TokenPAN, avviene in un processo immediatamente successivo all'elaborazione delle transazioni, e fa utilizzo della lista di Bin Range e della lista di HTokenPAN, messe a disposizione da parte del TKM, come indicato nello UC TK8.

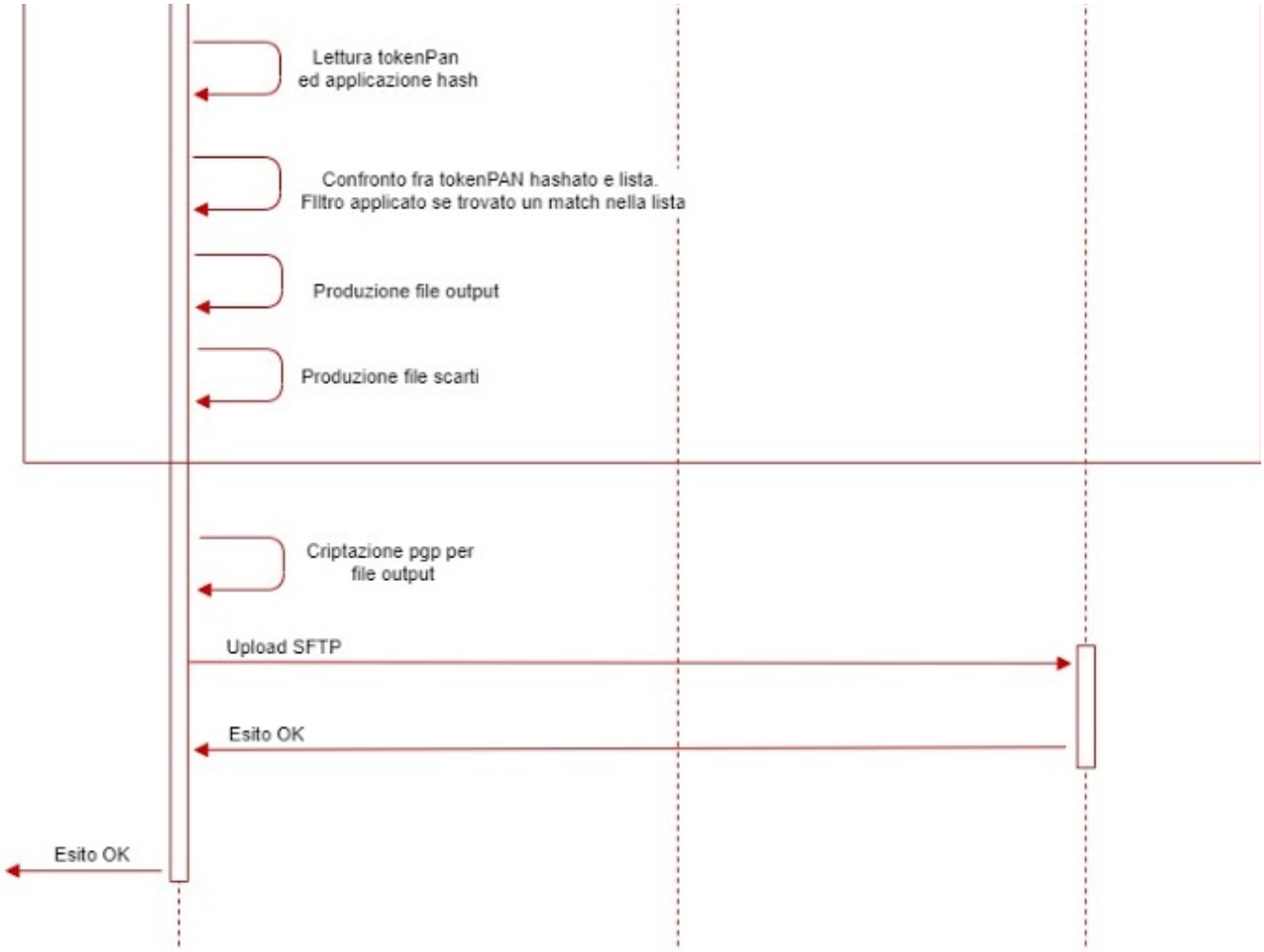
Il processo di elaborazione della lista di TokenPAN da inviare al TKM è composto dalle seguenti fasi:

- La Piattaforma PM (Token Manager) genera un flusso contenente una lista di Bin Range, contenente tutti gli intervalli di PAN valid strumenti enrollati al servizio CentroStella (vedere UC TK8)
- L'Acquirer, tramite la fase precedente del processo batch fornito, a tramite un processo propietati consolida i dati relativi a tutte le associazioni fra TokenPAN e PAR a sua disposizione, nel formato riportato nelle specifiche tracciato, e verrà depositato nella folder su cui è in polling il batch. La presenza del file è il trigger che fa partire il processo di elaborazione del batch.
- Il Batch invoca il servizio esposto dalla Piattaforma tramite il quale viene generato un link one shot, attivo temporaneamente per effettuare il download del della lista Bin Range, messa a disposizione del TKM
- Il Batch legge entrambi i flussi in input (lista Bin Range e lista Token) e, per ogni riga del file dei TokenPAN/PAR confronta il PAN con la lista di Bin Range. Un PAN è valido solo se presente all'interno di uno dei range forniti nella lista, definiti da due PAN interi, corrispondenti all'inizio e la fine del range. **Se presente nella lista il TokenPAN, Codice del circuito e PAR saranno validati ulteriormente, altrimenti verranno scartati una volta confrontati con tutti i bin range. Se il codice circuito è corrispondente al circuito Mastercard, i tokenPAN corrispondenti saranno trattati sempre trattati come validi in questa fase**.
- Il Batch invoca il servizio esposto dalla Piattaforma tramite il quale viene generato un link one shot, attivo temporaneamente per effettuare il download del della lista di HTokenPAN, messa a disposizione del TKM
- Il Batch confronta le transazioni filtrate per Bin Range con la lista di HTokenPAN del TKM, applicando l'hashing dei tokenPan (il salt utilizzato è lo stesso recuperato nella fase di elaborazione delle transazioni). **I tokenPan che saranno riportati nel tracciato di output sono quelle per cui l'hash del tokenPan non ha corrispondenza all'interno della lista prodotta dal TKM. Un record per cui è recuperato un match sarà definitivamente scartato, altrimenti sarà confrontato fintanto che in tutti i file contenenti gli HTokenPAN.**
- Il Batch, conclusa l'elaborazione al punto precedente, termina la scrittura del flusso filtrato in output, ed elimina tutti i dati ricevuti in input.
- Il Batch effettua la cifratura PGP del flusso di output.
- Il Batch invia il flusso dei TokenPAN al TKM, attraverso comunicazione sul canale sFTP utilizzato dal TKM (Nota: il canale sarà lo stesso impiegato per l'invio dei tracciati contenenti le transazioni da inviare a CentroStella)

Sequence Diagram




GET - /binrange/link
recupero lista file Bin Range
Repeat
GET - /download/file/{id}/{value}
download file Bin Range
GET - /htoken/known/link
recupero lista file HTokenPAN
Repeat
GET /download/file/{id}/{value}
download file HTokenPAN
Repeat
Salvataggio
Bin Range in memoria
Lettura tokenPan
ed estrazione Bin
Confronto fra primi quattro
caratteri PAN e lista Bin Range
Produzione file output parziale
Produzione file scarti
Repeat
Salvataggio
HTokenPan in memoria



Gestione errori

Il processo batch per la gestione delle transazioni deve prevedere, per una serie di possibili errori nel corso dell'elaborazione, una gestione opportuna del caso d'errore. Il processo nel suo insieme può essere diviso nelle parti riguardanti la produzione del tracciato delle transazioni da inviare a CentroStella, ed il susseguente processo per estrarre i tokenPAN da inviare al TKM. **Gli errori bloccanti per il processo di produzione dei tracciati del TKM non influenza la produzione dei tracciati di output relativi alle transazioni**.

Errori su chiamate per recupero lista PAR e PAN

Le chiamate REST ai servizi di recupero della lista HPAN/HTokenPAN, lista PAR, e per il salt da impiegare nell'hashing dei PAN delle transazioni possono produrre errori di connettività/timeout. Nel caso di errori nella fase di recupero di quest'informazione il processo viene interrotto, e dovrà essere ripetuta l'elaborazione sui flussi da filtrare. Nel caso una delle liste sia stata correttamente recuperata, il suo contenuto può essere mantenuto per la successiva esecuzione, senza la necessità di essere nuovamente recuperato.

Errori validazione file transazioni

I file recuperati dai servizi prevedono una validazione del contenuto, nella sua forma compressa, tramite validazione della checksum in SHA256. Ad entrambi i file sarà inoltre associato un campo che indica la data di generazione/modifica, per rendere possibile il controllo che il file ottenuto sia stato prodotto nella giornata di elaborazione. Nel caso la validazione del contenuto di una delle liste, sia per il checksum, che nel caso uno dei file non sia stato prodotto nella giornata, l'elaborazione deve essere ripetuta. Se uno dei due file recuperati nel corso della prima elaborazione ha completato correttamente il processo di validazione può essere mantenuto nelle successive esecuzioni giornaliere.

Errori di validazione su record

I tracciati relativi a singole transazioni che non sono validate secondo il formato previsto (Indicato nel paragrafo dedicato) non devono essere incluse nel file prodotto per l'invio a CentroStella. L'elaborazione del tracciato può essere mantenuta valida in caso di presenza di record con errori di validazione, se questi sono riportati in un tracciato dedicato, eventualmente da risottomettere ad elaborazione una volta regolarizzati i campi che hanno prodotto un errore di validazione.

Errori criptazione PGP

Errori nel processo di criptazione di un file di transazioni correttamente generato per l'invio in CentroStella potranno essere criptati senza dover necessariamente processare nuovamente il flusso originario. Il processo batch fornito da PagoPA dovrà essere invocato sono nel caso di risottomissione del flusso originale.

Errori invio su canale SFTP

I file criptati con chiave pubblica pgp che non siano correttamente inviati sul canale SFTP predisposto per l'invio dei flussi di transazioni verso CentroStella possono essere risottomessi senza necessariamente prevedere una nuova elaborazione del flusso originale. Il processo batch fornito da pagoPA potrà risottomettere tutti i file presenti in cartella di output.

Errori su chiamate

Le chiamate al servizi di recupero della lista Bin Range da impiegare nell'alborazione può produrre errori di connettività/timeout. Nel caso di errori nella fase di recupero di quest'informazione il processo viene interrotto, e dovrà essere eseguita una nuova elaborazione una volta ottenuto correttamente la lista fornita da TKM.

Errori validazione file tokenPan

I file recuperati dai servizi prevedono una validazione del contenuto, nella sua forma compressa, tramite validazione della checksum in SHA256. Ad entrambi i file sarà inoltre associato un campo che indica la data di generazione/modifica, per rendere possibile il controllo che il file ottenuto sia stato prodotto nella giornata di elaborazione. Nel caso la validazione del contenuto di una delle liste, sia per il checksum, che nel caso uno dei file non sia stato prodotto nella giornata, l'elaborazione deve essere ripetuta. Se uno dei due file recuperati nel corso della prima elaborazione ha completato correttamente il processo di validazione può essere mantenuto nelle successive esecuzioni giornaliere.

Errori di validazione su record

I tracciati relativi a singoli tokenPAN/PAR che non sono validati secondo il formato previsto (Indicato nel paragrafo dedicato) non devono essere incluse nel file prodotto per l'invio a TKM. L'elaborazione del tracciato può essere mantenuta valida in caso di presenza di record con errori di validazione, se questi sono riportati in un tracciato dedicato, eventualmente da risottomettere ad elaborazione una volta regolarizzati i campi che hanno prodotto un errore di validazione.

Errori criptazione PGP

Errori nel processo di criptazione di un file di tokenPAN correttamente generato per l'invio a TKM potranno essere criptati senza dover necessariamente processare nuovamente il flusso originario. Il processo batch fornito da PagoPA dovrà essere invocato sono nel caso di risottomissione del flusso originale.

Errori invio a TKM

I file criptati con chiave pubblica pgp che non siano correttamente inviati sul canale predisposto per l'invio dei tracciati verso TKM potranno essere inviati senza passare nuovamente dall'elaborazione del tracciato originale. Il processo batch fornito da PagoPA potrà inviare i tracciati presenti nella folder di output, in caso di nuove elaborazioni.

Formato del tracciato

Di seguito vengono descritti i dettagli relativi al flusso da impiegare nel processo per identificare quali Token dovranno essere trasmessi a PagoPA (Token Manager) per salvare le eventuali associazioni fra Token e strumenti di pagamento enrollati nel Payment Manager. **Il seguente tracciato viene prodotto internamente al processo batch, tramite l'estrazione dei tokenPAN che per cui esiste un match nella lista di Bin Range (fatta eccezione per il circuito Mastercard, dove sarà sempre considerata valida l'estrazione per Bin Range), e che non abbiano un riscontro fra hash del token e la lista di HTokenPAN del TKM**.

La naming convention del file è la seguente:

[servizio].[ABI].[tipofile].[data].[ora].[nnn].csv

in particolare:

servizio: fisso a 'TKM' (3 digit alfanumerico)

ABI: ABI del mittente (5 digit numerico)

tipofile: fisso a tipologia di file (6 digit alfanumerico)

nnn: progressivo file (3 digit numerico)

Campo	Formato	Note
servizio	Alfanumerico - 3 char	valore fisso TKM

ABI	Alfanumerico - 5 char	codice ABI del mittente
tipo_file	Alfanumerico - 6 char	tipologia del flusso inviato. Valore fisso a TKNLST
[data].[ora]	YYYYMMDD.HHMISS	timestamp di creazione del file
nnn	Alfanumerico - 3 char	Valore progressivo del file (es. 001)

Si precisa che:

Il file è in formato .csv, con separatori “;”

Il contenuto del file non prevede record di testa e coda ma solo record di dettaglio, secondo questo tracciato:

Campo	Tipo	Obbligatorio	Note
TokenPAN	Alfanumerico	SI	TokenPAN (in chiaro)
Circuito	Alfanumerico	SI	Circuito associato alla carta, informazione proveniente dal flusso di acquiring
PAR	Alfanumerico	SI	PAR del TokenPAN

Indicazioni per soluzioni proprietarie

Gli Acquirer che hanno in precedenza sviluppato una loro soluzione interna per l’elaborazione ed invio dei flussi di transazioni filtrati da strumenti non attivi in CentroStella dovrà intervenire, volendo mantenere la soluzione proprietaria, come segue:

- Introduzione di chiamata per recupero della lista PAR, similmente a quanto effettuato per recupero del salt e lista HPAN
- Modifica del tracciato input/output per aggiungere in coda il campo PAR (O solo il separatore per indicare campo vuoto, dove non siano presenti questi valori)
- Modificare il controllo effettuato per confrontare i PAR con la nuova lista ottenuta. **Una transazione è valida se almeno una delle due condizioni fra il vecchio controllo e quello nuovo è verificata**.

Il processo per la generazione della lista TokenPAN dovrà essere dipendente dalle transazioni dal processo di recupero delle transazioni, in quanto dovranno essere inviate le associazioni tokenPAN/PAR delle transazioni scartate nella fase di produzione del tracciato già in utilizzo, per cui esista una corrispondenza dei tokenPAN all’interno di uno dei range della lista Bin Range. Nel caso in cui si preferisca avere una soluzione proprietaria al posto del processo batch predisposto da PagoPA per gestire questo caso, dovrà essere predisposta una sezione completamente nuova, che preveda quanto segue:

- Gestione dei tracciati di input contenenti TokenPAN, codice circuito e PAR delle transazioni scartate nel corso della produzione del flusso da inviare a CentroStella
- Recupero della lista Bin Range tramite servizio dedicato. Il meccanismo di recupero prevede una prima chiamata per ottenere la lista di file da scaricare, su cui effettuare delle chiamate in GET per il recupero dei file.
- Confronto del tracciato contenenti le transazioni scartate nella prima elaborazione con la lista di Bin Range. Una transazione è considerata valida se il codice circuito è 02 (Mastercard), oppure il PAN (in chiaro), è contenuto all’interno di uno dei range indicati nella lista. Una transazione è da scartare solo dopo aver confrontato i valori con tutti i range disponibili.
- Recupero della lista di HTokenPAN tramite servizio dedicato. Il meccanismo di recupero prevede una prima chiamata per ottenere la lista di file da scaricare, su cui effettuare delle chiamate in GET per il recupero dei file.
- Confronto delle transazioni validate rispetto al Bin Range, con la lista di HTokenPAN nella seconda lista ottenuta. I Pan dovranno essere hashati utilizzando il salt recuperato nella prima elaborazione, una transazione è considerata da scartare se ESISTE un match all’interno di uno dei file. Se una transazione non ha un HTokenPAN all’interno dei file, è considerata valida, e verrà riportata nel file di output.
- Applicazione criptazione PGP con chiave pubblica fornita per il servizio
- Il Batch invia il flusso dei TokenPAN al TKM, attraverso comunicazione sul canale sFTP utilizzato dal TKM (Nota: il canale sarà lo stesso impiegato per l’invio dei tracciati contenenti le transazioni da inviare a CentroStella)

TK11. Flusso delle transazioni: adattamento delle logiche di match e salvataggio nuove associazioni

TK11 - Flusso delle transazioni: adattamento delle logiche di match e salvataggio nuove associazioni	
Attori Abilitati	Centro Stella (RTD/BPD), TKM
Funzionalità	• elaborazione delle transazioni attraverso la nuova logica di matching HPAN/HToken, PAR • Salvataggio a DB delle nuove associazioni HToken-PAR ed allineamento del TKM

Assunti e Punti di attenzione	• Ai fini della elaborazione delle transazioni non sarà necessario fare un controllo sulla data di consenso all'utilizzo delle carte tokenizzate. Non sarà dunque invocare il servizio della GetStatus del TKM. • Per gli enrollment Token da Issuer la data di attivazione/disattivazione tokenizzate verrà generata dal TKM in quanto owner del dato sul consenso cittadino all'utilizzo PAR. Anche in caso di nuove associazioni rilevate dal TKM e propagate verso il Centro Stella (PAR e HToken) la data di enrollment verrà generata dal TKM. In questo modo si eviteranno eventuali disallineamenti dovuti al processo asincrono, tra l'orario effettivo di conferma del consenso da parte del Cittadino e l'orario generato dal sistema. • Si precisa che il file delle transazioni inviato su CentroStella conterrà solo record afferenti a carte enrollate.

Contesto

Per la corretta elaborazione delle transazioni afferenti a strumenti tokenizzati, nella soluzione proposta è prevista l'integrazione del campo PAR nel tracciato inviato dagli Acquirer verso il Centro Stella. Il PAR sarà impostato come campo opzionale per gestire le differenti tempistiche di integrazione dei vari Acquirer. Inoltre, come riportato anche nell'assunto #6, i 3 circuiti principali si comportano in modo differente e in certi casi il PAR potrà essere reso disponibile solo dopo la generazione del primo tokenPAN.

Secondo quanto descritto nei paragrafi successivi grazie alla presenza del PAR sarà possibile rafforzare la logica di matching delle transazioni in input al fine di poter riconoscere e premiare le operazioni afferenti a carte tokenizzate.

Il Centro Stella potrà ricevere il PAR attraverso i seguenti attori:

- tramite Issuer. Si precisa he tutti i token enrollati insieme al PAR o successivamente tramite patch avranno la stessa data di enrollment del PAR (le funzionalità di enrollment/patch saranno introdotte già con la soluzione Q&D ma saranno condizione necessaria per il corretto funzionamento delle tokenizzate target)
- tramite TKM il quale invierà o solo PAR recuperati dai circuiti oppure l'associazione HToken/PAR per via delle nuove coppie rilevate in fase di elaborazione delle lista TokenPAN inviata dagli Acquirer. Anche in questo caso la data di enrolment del PAR verrà propagata su tutti gli Htoken afferenti a quel PAR.

Processo

Il workflow di elaborazione si scatena all'atto della ricezione di una nuova transazione su RTD inviata tramite batch Acquirer e sarà composto da due sotto-processi:

1. Sottoprocesso di ricezione e premiazione transazioni

- Il sistema SIA riceve in input i flussi batch in formato standard RTD provenienti dai diversi Acquirer. Tramite un processo batch, il file contenuto nella NAS verrà processato al fine di leggere il record.
  Ogni record elaborato viene successivamente scritto su una coda per essere inviato al servizio RTD.
- Tramite una coda, i tracciati vengono inviati al componente RTD rilasciato sul Cloud Azure (BPD), per essere sottoposti ai seguenti step. Tramite un gestore delle transazioni verrà effettuato un primo filtro sul record al fine di verificare l'esistenza dei seguenti vincoli:
  - l*a transazione dovrà pervenire sulla piattaforma all'interno di un intervallo temporale in cui risulti attivo un periodo di erogazione* come previsto dal decreto ministeriale di riferimento;
  - lo strumento di pagamento deve essere in stato attivo: l'HPAN (carta master nel caso delle tokenizzate) dev'essere attiva alla data della transazione.
  - *la transazione per essere premiabile deve risultare associabile ad uno strumento di pagamento che risulti attivo su BPD nel momento in cui è stata effettuata la transazione. La nuova logica di matching sarà determinata da due parametri: HPAN per transazioni effettuate con carta padre o PAR, parametro di riferimento utile per effettuare il controllo sullo stato delle carte tokenizzate.* Pertanto si renderà dunque necessario verificare con quale ID strumento è stata effettuata l'operazione (con carta tokenizzata o meno).
    - se nella transazione sarà valorizzato un **HPAN** il controllo verrà effettuato rispetto alla data di enrolment dello stesso ed in caso risulti antecedente alla data della transazione, quest'ultima potrà essere candidata al cashback.
    - nel caso di **HToken** valorizzato nel tracciato e presente nel database, il controllo si farà confrontando la data meno recente tra l'enrolment del PAR associato e enrolment date del Token stesso con la data della transazione. Se la transazione ha una transaction date maggiore verrà premiata. Anche in questo caso se il controllo sulle date risulta idoneo allora la transazione potrà essere premiabile.
    - nel caso di **HToken** valorizzato nel tracciato, ma non presente nel database, sarà possibile premiare la transazione basandosi sulla data di attivazione PAR. L'assunto è che l'associazione tra PAR/HToken e PAR/HPAN è univoca.

Si specifica che, in mancanza di almeno uno dei requisiti sopra elencati, la transazione verrà scartata. Per maggiori dettagli sulle logiche di scarto si rimanda al paragrafo **"Gestione casi di errore ed eccezioni".**

- Una volta appurati i punti sopra, si rende necessario verificare a quale tipologia di operazione appartiene la transazione. **In presenza di storno sarà effettuato un ulteriore controllo al fine di verificare se quest'ultimo risulta correlato ad una transazione originaria.** S olo in presenza di tale correlazione è possibile constatare che lo strumento di pagamento era già enrollato alla data della transazione originaria. Tale evidenza giustifica la variazione negativa di cashback legata allo storno, pertanto l'utente sarà in grado di visualizzare da APP IO entrambe le transazioni. In mancanza di tale correlazione, invece, la transazione non verrà sottoposta alle logiche di calcolo applicate da BPD al fine di concorrere alla generazione del cashback. Lo storno non correlato verrà salvato a DB ma non verrà mostrato

nel servizio di “recupero lista transazioni” restituito all’APP IO. Si specifica che ***l’algoritmo di matching utile per legare uno storno alla propria transazione originale, rimarrà invariato anche in presen pagamento con carte tokeinzzate.***

2. Sotto processo di aggiornamento della base dati di BPD ed allineamento del TKM

Per tutte le transazioni premiate (non scartate), è possibile procedere con un processo automatico di allineamento della base dati interna di BPD attraverso la quale sarà possibile salvare tutti i nuovi Htoken inviati nel flusso standard pagoPA e collegati ad un PAR enrollato sul servizio.

Successivamente, al fine di mantenere i sistemi allineati, BPD scriverà su una coda condivisa con il TKM tutte le nuove associazioni HToken, PAR ed HPAN.

Ciò anticipato si distinguono le seguenti casistiche:

Caso 1. HToken non presente nel DB, il PAR è presente

In questo caso avendo il PAR (univoco per la carta fisica) l’Htoken verrà salvato nel database come figlio del HPAN master. L’associazione viene scritta dal Centro Stella su una coda letta da TKM per allineare l’anagrafica senza invocare i circuiti. BPD salverà a DB il token come figlio del HPAN padre con data di enrollment pari a quella dell’attivazione PAR associato.

Caso 2: HPAN è presente nel database, mentre manca il PAR

Il PAR potrebbe non essere presente nel database perché non è stato ancora rilevato dai processi di recupero di TKM oppure perchè non è stato dato il consenso di utilizzare le tokenizzate da parte del Cittadino.

In questo caso, dopo aver verificato che lo strumento sia effettivamente un HPAN, la piattaforma BPD non salverà l’informazione del PAR nel database e non verrà invocata nessuna chiamata aggiuntiva verso il TKM come da assunto.

Caso 3: HPAN è presente nel database e il PAR è presente

In questo caso, non sarà necessario effettuare nessun altra operazione aggiuntiva

Caso 4: HToken presente nel database, il PAR è mancante

Non potrebbe verificarsi in quanto l’associazione PAR-Token verrà inviata e salvata esclusivamente a seguito della ricezione dell’associazione dal TKM.

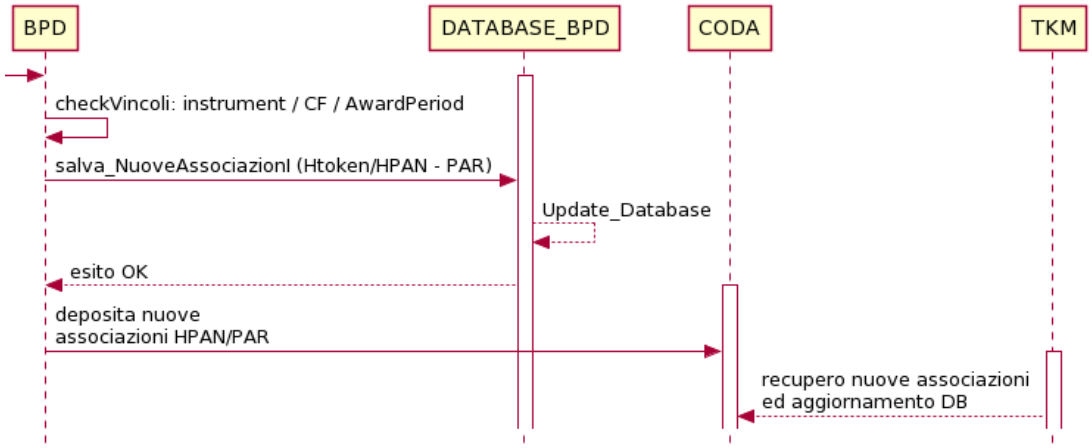
Comunicazione BPD-TKM tramite coda:

In merito alla struttura del messaggio da BPD verso TKM, il messaggio conterrà i seguenti dati necessari:

CF; PAR; HPAN; array HToken

- il messaggio dovrà essere ***cifrato*** con chiave pgp
- verrà valutata la possibilità di condividere con il TKM una lista di header per identificare i diversi messaggi in modo da poter gestire correttamente i vari processi, tra cui anche il processo in questione
- Assunto: Il TKM dovrà gestire le informazioni hashate dei vari strumenti

Sequence Diagram



Logiche di filtro delle transazioni secondo la data di registrazione al servizio

Caso	Data di disattivazione	Data contabile transazione *	Esito

I	-	dTR = dATT	Accettata
II	-	dTR < dATT	Scartata
III	-	dTR > dATT	Accettata
IV	dDIS = dATT	dTR = dDIS	Scartata
V	dDIS = dATT	dTR > dDIS	Scartata
VI	dDIS = dATT	dTR < dDIS	Scartata
VII	dDIS > dATT	dTR > dDIS	Scartata
VIII	dDIS > dATT	dATT <= dTR > dDIS	Accettata

Qualsiasi fascia oraria nella data contabile (dATT = data di attivazione, dDIS = data di disattivazione, dTR = data transazione)

Gestione casi di errore ed eccezioni

01 - Strumento non attivo

- Si prevede la gestione dell'eccezione del caso in cui a seguito di un check effettuato sullo strumento di pagamento, quest'ultimo o il PAR associato ad esso risulti non attivo

02 - Periodo non attivo

- Nel caso in cui a seguito di un check effettuato sulla data di transazione risulti che la data non rientra in nel periodo attivo, la transazione verrà scartata.

03 - CF non attivo

- Se dal controllo dovesse risultare che l'utente non ha accettato i T&C, la transazione non sarà premiata e verrà quindi scartata.

04 - Generic Error

- Sono previsti degli errori generici in caso di errori di comunicazione con la coda del TKM oppure in caso di problemi di connessione DB.

GESTIONE ENROLLMENT, AGGIORNAMENTO E DISATTIVAZIONE TOKENIZZATE

Per completezza, si riportano di seguito i processi di enrolment carta tokenizzata, di aggiornamento Token e di disattivazione carta con token associati. Questi processi saranno esposti brevemente nei successivi Use Case in modo da descrivere l'intero ciclo di vita di una carta tokenizzata.

TK.12 Enrollment Carta Tokenizzata da H/M Banking

Overview Generale

La funzionalità in oggetto permette all'utente di abilitare uno strumento di pagamento tokenizzato al servizio Cashback tramite portale Issuer.

In fase di enrollment della carta tokenizzata da Issuer, oltre al PAN dello strumento di pagamento, sarà possibile ricevere, quando disponibili, anche i token PAN associati allo stesso ed il PAR (facoltativo). In presenza di carte tokenizzate, l'enrollment si basa su una logica uno a molti (1: M), ovvero, all'interno di una singola chiamata di attivazione al servizio, verranno forniti più identificativi token.

I parametri del PAN e lista Token verranno inviati al PM, il quale restituirà a BPD, solo per la carta fisica il corrispondente valore hashato. BPD salverà nel proprio DB esclusivamente il dato dell'hpan enrollato su DB generando anche il timestamp di attivazione carta fisica.

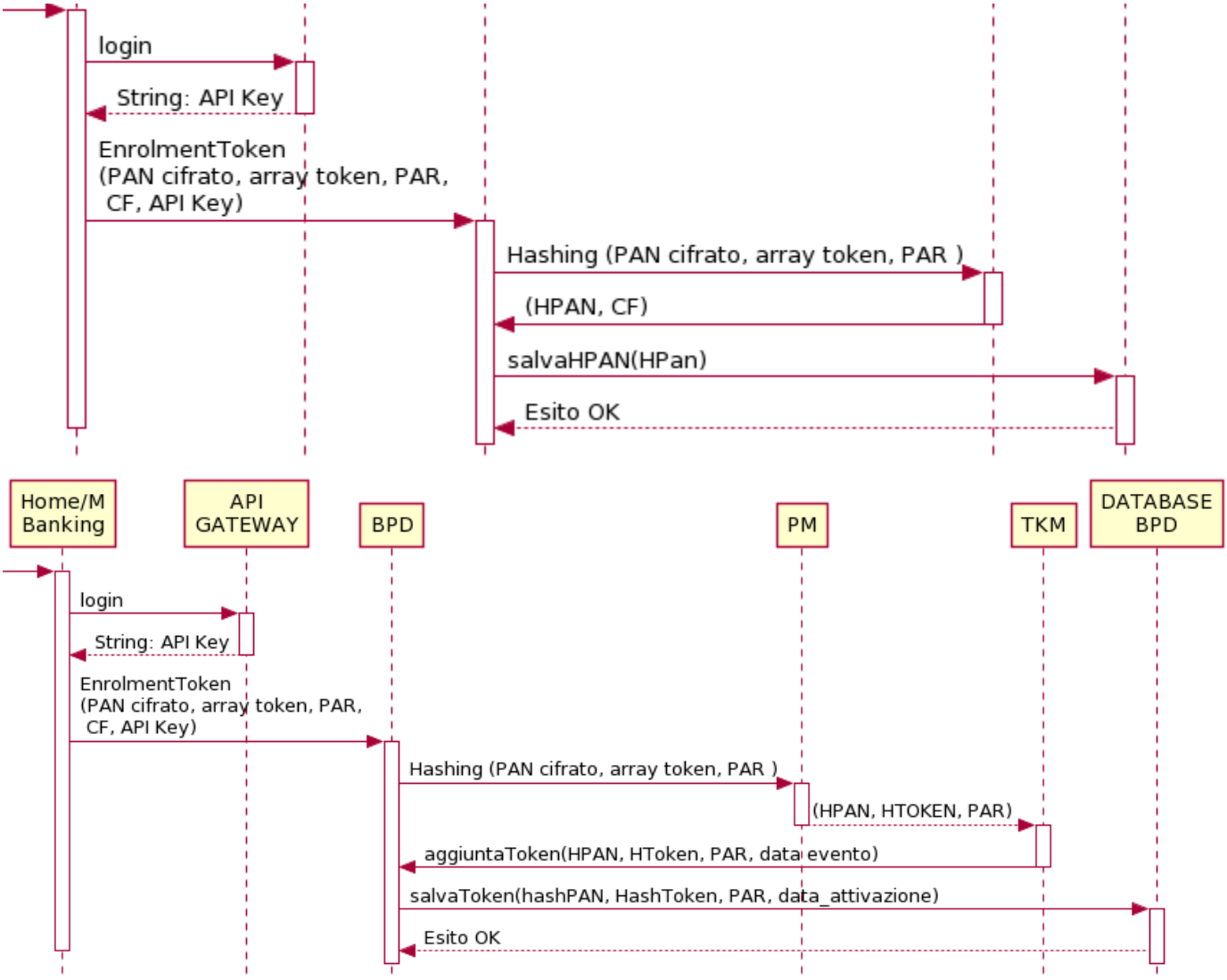
Nessuna informazione relativa ai HToken non verrà salvata su BPD in questa fase del processo.

Oltre ad effettuare il hashing e restituire il valore del Hpan a CentroStella, il PM veicolerà successivamente i dati hashati (sia dell'HPAN che degli HToken) verso il TKM. Quest'ultimo verificherà se per la carta oggetto di richiesta è stato fornito il consenso all'utilizzo Token.

Si precisa quindi che i token saranno oggetto di hashing dal PM, ma non verranno restituiti a BPD come per la carta fisica. Come da assunto, l' associazione HToken/Hpan/PAR verrà salvata successivamente al momento della ricezione legame da parte del TKM.

Per tutti gli HToken ricevuti dal TKM verrà salvata l'informazione dell'HPAN della carta fisica associata e del PAR, in modo da tracciare correttamente la relazione tra il PAR, PAN della carta fisica e i relativi token secondo le regole descritte nella tabella sopra.

Sequence Diagram



TK.13 Aggiornamento Token dal H/M Banking

Overview Generale

La funzionalità in oggetto permetterà l'invio verso CentroStella la lista token aggiornati associati ad uno strumento di pagamento tokenizzato.

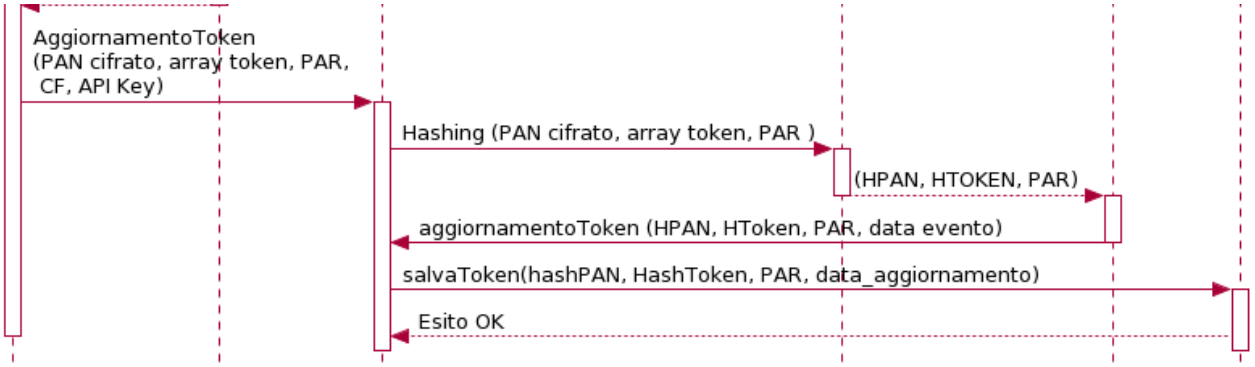
La piattaforma BPD metterà a disposizione il servizio che potrà essere invocato dai touchpoint degli Issuer, per consentire l'aggiornamento dei tokenPAN relativi a carte (HPAN) precedentemente registrate su BPD da un utente.

All'atto della richiesta di aggiornamento (aggiunta/rimozione token), l'Issuer invierà una "fotografia" di tutti i token associati al PAN della carta, attivi e validi al momento della chiamata. A seguito della ricezione chiamata, BPD invierà i dati al Payment Manager per effettuare l'hashing del PAN e dei Token associati. Il PM veicolerà l'informazione ricevuta verso il TKM per allineare quest'ultimo, mentre non restituirà nessun dato hashato al CentroStella.

In questa fase non è previsto nessun salvataggio dati da parte di BPD.

Il TKM a seguito della richiesta di aggiornamento ricevuta dal PM, procederà con le verifiche e invierà il set aggiornato di HToken/HPAN e PAR a BPD. BPD salverà i dati secondo le regole indicate nella tabella esposta nel TK.9.

Sequence Diagram



TK.14 Disattivazione Carta tokenizzata dal H/M Banking

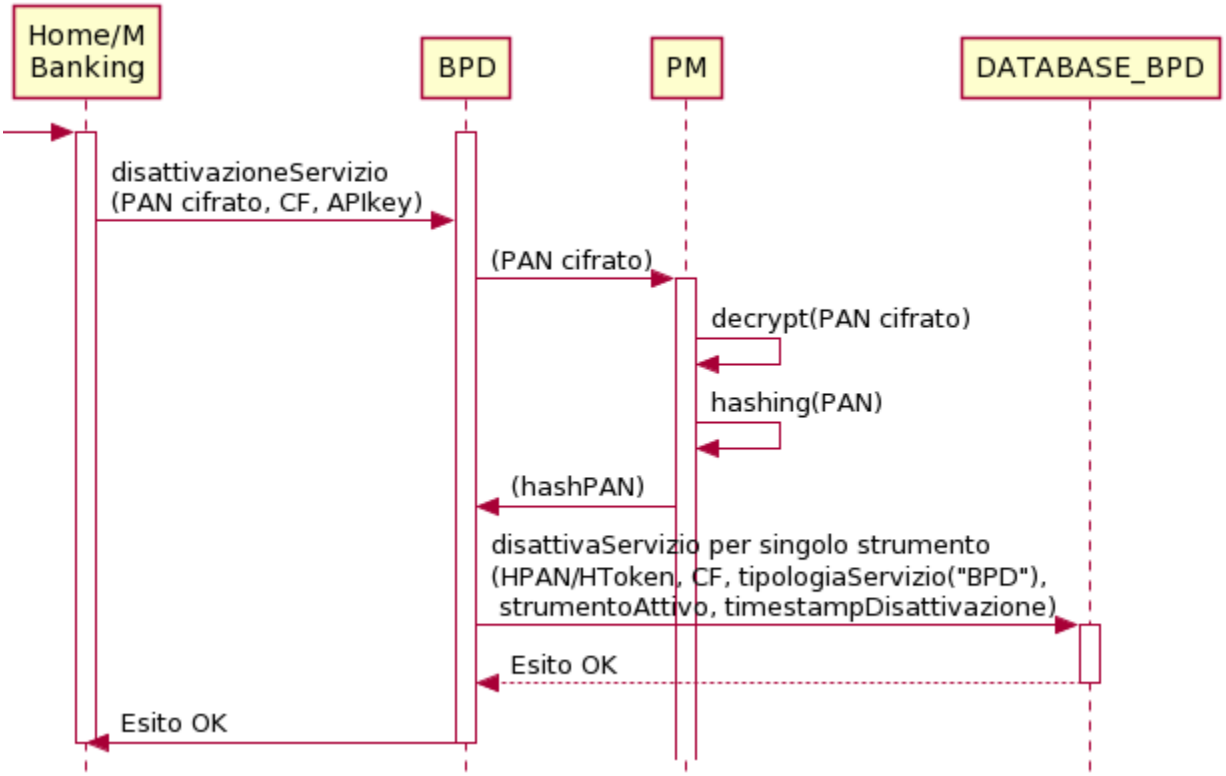
Overview generale

In fase di disattivazione della carta fisica, in presenza di carte tokenizzate, la disattivazione si basa su una logica uno a molti (1:M), ovvero, a seguito di una singola chiamata proveniente dal client contenente solo l'identificativo della carta fisica, verranno disattivati tanti strumenti, quanti sono gli identificativi univoci delle carte (PAN e Token) presenti nel database.

L'utente seleziona dal H/M Banking l'opzione di richiesta disattivazione servizio dallo strumento di pagamento e il H/M Banking invierà verso la piattaforma BPD la chiamata di disattivazione servizio BPD inviando il dato del numero di carta criptato. BPD veicola verso il PM il dato dello strumento di pagamento inviato il quale decifra il dato del PAN e invia il valore hashato a BPD che procede con la disattivazione del servizio dallo strumento indicato e tutti gli eventuali token associati.

Si precisa che a seguito della richiesta di disattivazione della carta fisica verrà aggiornato lo stato di ogni singolo strumento (hpan e htoken) in Inactive indicando anche il timestamp della disattivazione (il campo verrà ricavato, al momento della chiamata, dai servizi esposti da BPD)

Sequence Diagram



Design di dettaglio

Componente RTD - Payment Instrument Manager

Responsabilità

Il componente ha la responsabilità di gestire il recupero delle informazioni sugli strumenti di pagamento enrollati in CentroStella, per i programmi BPD ed FA, da utilizzare per generare le liste contenenti rispettivamente le lista di HPAN/TokenPAN e PAR.

Il componente deve trattare il recupero delle informazioni, e la susseguente generazione, ed esposizione dei file prodotti, entro i tempi utili per rendere disponibile giornalmente i flussi.

Il componente deve gestire i volumi, derivate dalla lettura giornaliera dell'intero contenuto della lista di strumenti attivi.

Dipendenze

Il componente deve dialogare con il servizio Azure Blob Storage, mediante l'utilizzo delle API ufficiali del servizio, per poter caricare i file generati, ed ottenere dei link one-shot per il download.

Il componente s'interfaccia con le basi dati dei componenti di BPD ed FA per ottenere tutte le informazioni necessarie per aggiornare la base dati RTD con i dati degli strumenti di pagamento attivi.

I componenti di BPD che sono coinvolti sono:

- Componente BPD Award Period -> Recupero delle informazioni per il periodo di premiazione corrente
- Componente BPD Citizen -> Recupero delle informazioni sui Citizen non più iscritti al servizio, per gestire la cancellazione degli strumenti associati
- Componente BPD Payment Instrument -> Recupero delle informazioni relative agli strumenti iscritti a BPD da inserire, cancellare, o aggiornare per associazione del PAR

I componenti di FA che sono coinvolti sono:

- Componente FA Payment Instrument –> Recupero delle informazioni relative agli strumenti di pagamento iscritti ad FA da inserire, cancellare, o aggiornare per associazione del PAR

Il componente s'interfaccia con il database predisposto per RTD, dove vengono gestiti i dati degli strumenti di pagamento da includere nelle liste, e le informazioni sulle esecuzioni per l'estrazione in delta dei dati sugli strumenti di pagamento.

Il componente viene contattato, intermediato dall'API Manager di Azure, dai processi degli acquirer, per ottenere il link da cui scaricare i file contenenti le liste HPAN e PAR.

Data-layer

Entità logiche

Il componente gestisce un'entità che definisce le informazioni necessarie per definire gli strumenti attivi sugli ambienti BPD/FA:

- hpan String, corrisponde all'hash del Pan/TokenPAN degli strumenti di pagamento
- bpd_enabled Booleano, indica la condizione dello strumento (attivo o meno) su BD
- fa_enabled Booleano, indica la condizione dello strumento (attivo o meno) su FA
- insert_date Timestamp, indica la data d'inserimento in tabella RTD
- insert_user String, corrisponde all'utenza che ha effettuato l'inserimento in tabella RTD
- update_date Timestamp, indica la data di ultimo aggiornamento per lo strumento di pagamento
- update_user String, corrisponde all'utenza che ha effettuato l'aggiornamento in tabella RTD
- par String, dato per l'associazione di un token ad una carta fisica, se presente

L'entità indica uno strumento di pagamento attivo, se almeno uno dei due flag è a settato a TRUE. Le informazioni che saranno impegnati nelle liste saranno l'HPAN/HTokenPAN, ed il PAR.

Il componente gestisce un'entità che viene utilizzata per definire il delta da applicare al processo di estrazione, che contiene:

- Data inserimenti BPD Timestamp, indica l'ultima data di estrazione da BPD per inserimento/aggiornamento degli strumenti di pagamento enrollati al servizio
- Data cancellazione BPD Timestamp, indica l'ultima data di estrazione da BPD per la disabilitazione degli strumenti di pagamento enrollati al servizio
- Data inserimenti FA Timestamp, indica l'ultima data di estrazione da FA per inserimento/aggiornamento degli strumenti di pagamento enrollati al servizio
- Data cancellazione FA Timestamp, indica l'ultima data di estrazione da FAper la disabilitazione degli strumenti di pagamento enrollati al servizio

Write access-patterns

L’entità logica che racchiude lo stato delle informazioni sugli strumenti di pagamento, ottenuta tramite lettura dai database dei componenti di BPD ed FA, verranno trattate per operazioni d’inserimento, aggiornamento e cancellazione effettuata in bulk, di dimensione regolabile attraverso una configurazione del componente.

In fase di generazione del file vengono eseguite scritture in bulk su file temporanei, presenti sullo spazio di archiviazione su disco a disposizione del componente.

Read access-patterns

Il componente effettua letture da più database, per ottenere le diverse informazioni utili per aggiornare la lista degli strumenti attivi.

Le informazioni per cui sono eseguite operazioni di lettura singole sono:

- Date di ultima elaborazione dell’aggiornamento, estratte dalla tabella RTD
- Date di inizio e fine periodo di elaborazione corrente, estratte dal database predisposto per il componente BPD

Le informazioni per cui sono eseguite operazioni lettura in bulk, con dimensione configurabile a livello di applicazione, per singola estrazione:

- Citizen disabilitati dall’ultima esecuzione, ottenuta da lettura al database/schema corrispondente in BPD. Quest’informazione viene utilizzata per identificare strumenti di pagamento corrispondenti da disabilitare all’interno della tabella RTD
- Payment Instrument BPD da inserire/aggiornare, ottenuta da lettura al database/schema corrispondente in BPD
- Payment Instrument BPD da disabilitare, ottenuta da lettura al database/schema corrispondente in BPD
- Payment Instrument FA da inserire/aggiornare, ottenuta da lettura al database/schema corrispondente in FA
- Payment Instrument FA da disabilitare, ottenuta da lettura al database/schema corrispondente in FA

Data-store e modello di dominio

Le entità indicate nel paragrafo alle entità logiche sono salvate nelle tabelle predisposte nel database predisposto per il componente, le entità sono rispettivamente:

- *rtd_database.rtd_payment_instrument_data* Tabella che include l’entità logica contenente i dati dei singoli strumenti di pagamento da utilizzare per la produzione delle liste
- *rtd_database.rtd_exec_date* Tabella che include l’entità che memorizza i dati delle ultime esecuzioni

Monitoring

Il monitoraggio prevede il controllo degli errori applicativi del componente, tramite verifica dei log inviati su coda Kafka dedicata. Gli errori sono monitorati regolarmente con cicli di cinque minuti.

La generazione del file è regolarmente monitorata con verifica giornaliera, secondo cutoff indicato per il termine della produzione del file giornaliero, che prevede allarme con severity critica nel caso di mancata generazione di uno dei due file.

Modelli di *fault-tolerance*

Il componente, in caso l’elaborazione s’interrompa per motivi dovuti a mancata connessione ai database, o allo storage Azure, può essere ripetuta tramite chiamata al servizio di elaborazione, messo a disposizione tramite API. Lo stato delle tabelle verrà aggiornato a partire dall’ultima elaborazione correttamente conclusa.

Scalabilità

Il componente deve poter generare un file, trattando tutte le informazioni all’interno di un documento direttamente accessibile. Non è previsto che il componente abbia multiple repliche, e tutte le operazioni per trattare la scalabilità devono essere operate su base verticale, dando a disposizione del componente più risorse, dove necessario.

RTD Payment Instrument Manager API - Avvio aggiornamento liste HPAN/PAR

Interna	*Y*
Esterna	*N*
Sincrona / Asincrona	*A*
Autorizzazione	*N.A*

Autenticazione	*N.A*
TPS (stimato)	*N.A*
Idempotente	*Y*
Stateless	*Y*
Read / Write intensive	*R/W*
Cacheable	*N*

Signature

***Path**: /rtd/payment-instrument-manager/generate-active-lists*

***Method**: POST*

API tramite cui viene fatta richiesta di eseguire il processo di aggionamento ed upload delle liste degli HPAN e PAR, da impiegare nel processo batch degli acquirer per filtrare le transazioni. La chiamata fa partire un processo asincrono che si occuperà di:

- Aggiornare la tabella RTD con i dati ottenuti dall'estrazione in delta, ottenute da chiamate ai database BPD/FA
- Estrazione dati da RTD per generare le lista HPAN e PAR, processate con select paginate di dimensione fissa e configurabile
- Compressione dei file ed upload su Azure

La chiamata ritornerà esito subito dopo aver avviato il processo.

Errori

HTTP Response Code	Descrizione
500	Applicativo - Errore Generico

Request

Header

Nessun Header dedicato è previsto per la request

Body

Nessun body è previsto per la request

Response

Header

Nessun Header dedicato è previsto per la response

Body

Nessun body è previsto per la response

Monitoring

Il monitoraggio prevede il controllo degli errori applicativi del componente, tramite verifica dei log inviati su coda Kafka dedicata. Gli errori sono monitorati regolarmente con cicli di cinque minuti.

La generazione del file è regolarmente monitorata con verifica giornaliera, secondo cutoff indicato per il termine della produzione del file giornaliero, che prevede allarme con severity critica nel caso di mancata generazione di uno dei due file.

RTD Payment Instrument Manager API - Recupero lista PAR

Interna	*Y*

Esterna	*Y*
Sincrona / Asincrona	*S*
Autorizzazione	*Autorizzazione tramite chiave di sottoscrizione Azure*
Autenticazione	*Endpoint protetto da certificato SSL, fornito da security PagoPA*
TPS (stimato)	*TBD*
Idempotente	*Y*
Stateless	*Y*
Read / Write intensive	*N*
Cacheable	*N*

Signature

***Path**: /rtd/payment-instrument-manager/active-par-list*

***Method**: GET*

API tramite cui viene fatta richiesta della risorsa contenente la lista dei PAR, da impiegare nel processo batch degli acquirer per filtrare le transazioni sulla base del PAR. L'endpoint prevede l'utilizzo del meccanismo di recupero dell'informazione tramite una risposta con codice 302, contenente nella response il link da contattare per il recupero del file.

Errori

HTTP Response Code	Descrizione
404	Risorsa - File non trovato
500	Applicativo - Errore Generico
401	Errore di autorizzazione -Subscription key mancante o errata
403	Errore di autenticazione - Certificato SSL non valido

Request

Header

Campo	Tipo	Obbligatorio	Descrizione
Ocp-Apim-Subscription-Key	Alfanumerico	SI	Chiave di sottoscrizione per API Manager Azure

Response

Header

Campo	Tipo	Descrizione
Location	Alfanumerico	Campo standard per HTTP 302, contenente il link one-shot per il download della lista
x-ms-meta-sha256	Alfanumerico	Checksum SHA256 del file
last-modified	Timestamp	Timestamp corrispondente all'ultima data di aggiornamento del file

Monitoring

Il monitoraggio prevede il controllo degli errori applicativi del componente, tramite verifica dei log inviati su coda Kafka dedicata. Gli errori sono monitorati regolarmente con cicli di cinque minuti.

Componente Acquirer Batch

Responsabilità

Acquisizione transazioni

Il componente batch delle transazioni è il componente che viene messo a disposizione dei singoli acquirer, per facilitare il processo d' integrazione nella produzione del tracciato delle transazioni, filtrato dagli strumenti di pagamento non enrollati a CentroStella. Ha la responsabilità di centralizzare tutte le operazioni utili al fine di completare il processo di alimentazione in RTD dei tracciati.

Rispetto alla prima versione del processo batch, la nuova versione viene arricchita con la gestione delle informazioni relative alla presenza di carte tokenizzate, che vengono gestite tramite la precedente struttura dati della lista HPAN, che andrà a contenere anche l'hash delle TokenPAN, che della possibilità di riconoscere token di carte già enrollate, tramite l'introduzione del campo PAR nel tracciato delle transazioni, e della lista di PAR che verrà ottenuta dai servizi di CentroStella.

Il componente deve gestire i tracciati degli acquirer, e le informazioni pervenute da CentroStella su HPAN/TokenPAN e PAR, in modo da produrre dei flussi da sottoporre giornalmente entro i cutoff previsti. Il processo dovrà gestire inoltre l'elaborazione in modo da mantenere un uso delle risorse ottimale, a fronte dell'introduzione dell'insieme di dati relativi ai PAR.

Acquisizione tokenPAN

Il componente batch per l'elaborazione della lista TokenPAN, è il processo batch indipendente che viene messo a disposizione dei singoli acquirer, per facilitare il processo d'integrazione nella produzione del tracciato contenente le associazioni TokenPAN/PAR, da inviare al TKM. Ha la responsabilità di centralizzare tutte le operazioni utili al fine di completare il processo di alimentazione dei tracciati.

Rispetto alla prima versione del processo batch, la nuova versione viene arricchita con la gestione delle informazioni relative alla presenza di carte tokenizzate, che vengono gestite tramite la precedente struttura dati della lista HPAN, che andrà a contenere anche l'hash delle TokenPAN, che della possibilità di riconoscere token di carte già enrollate, tramite l'introduzione del campo PAR nel tracciato delle transazioni, e della lista di PAR che verrà ottenuta dai servizi di CentroStella.

Il componente deve gestire i tracciati, e le informazioni pervenute da CentroStella su HPAN/TokenPAN e PAR, in modo da produrre dei flussi da sottoporre giornalmente entro i cutoff previsti.

Dipendenze

Il componente è un processo batch eseguibile sugli ambienti dei soggetti acquirer, eseguito tramite avvio del bundle reso disponibile sul repository di CentroStella.

Acquisizione transazioni

Il componente s'interfaccia con gli ambienti degli acquirer tramite i file di tracciato dati come input al processo, ed opzionalmente può fare utilizzo di un database messo a disposizione dal soggetto acquirer per il salvataggio delle informazioni relativo alle esecuzioni del processo batch, costruito sul framework open source Spring Batch.

Il componente dovrà gestire il recupero delle informazioni relativa al salt per l'hashing dei PAN, la lista degli HPAN/HTokenPAN, e della lista PAR. Per ottenere queste informazioni il componente chiama i servizi messi a disposizione da CentroStella su Azure API Manager, legati al prodotto RTD, che permetteranno di ottenere i dati richiesti con endpoint protetti da sottoscrizione al servizio e mutua autenticazione con certificato SSL, firmato sulla base della CA di PagoPA.

Il componente può interfacciarsi con il canale sFTP per CentroStella, tramite cui saranno inviati i tracciati di output dell'elaborazione del batch.

Acquisizione tokenPAN

Il componente s'interfaccia con gli ambienti degli acquirer tramite i file contenenti i tokenPAN estratti dalle transazioni filtrate nel processo di filtro per invio a CentroStella, dati come input al processo di estrazione dei tokenPAN. Opzionalmente può fare utilizzo di un database messo a disposizione dal soggetto acquirer per il salvataggio delle informazioni relativo alle esecuzioni del processo batch, costruito sul framework open source Spring Batch.

Il componente dovrà gestire il recupero della seguenti liste:

- lista di Bin Range, per poter identificare i TokenPAN da trasmettere al TKM.
- lista tokenPAN relativa ad associazioni note al TKM, per cui non sarà necessario l'invio dell'informazione tramite il flusso prodotto dal processo di acquisizione

Per ottenere queste liste il componente chiama i servizi messi a disposizione su Azure API Manager, che permetteranno di ottenere i dati richiesti con endpoint protetto da sottoscrizione al servizio e mutua autenticazione con certificato SSL, firmato sulla base della CA di PagoPA.

Il componente può interfacciarsi con il canale sFTP definito per il TKM, tramite cui saranno inviati i tracciati prodotti dall'elaborazione del batch.

Data-layer

Entità logiche

Il processo batch gestisce, come principale entità logica del processo di elaborazione, la rappresentazione delle transazioni ottenute dalla lettura del tracciato, InboundTransaction, che si compongono di:

- idTrxAcquirer String, corrispondente al campo del tracciato *id_trx_acquirer*. Che rappresenta l'Identificativo univoco della transazione a livello di Acquirer.
- acquirerCode String, corrispondente al campo del tracciato *codice_acquirer*, che Codice ABI della banca Acquirer.
- trxDate Datetime, corrispondente al campo del tracciato *date_time*, che Timestamp dell'operazione di pagamento effettuata presso l' Esercente.
- pan String, corrispondente al campo del tracciato *hash_pan*, che rappresenta Hash del PAN/TokenPAN dello strumento di pagamento utilizzato.
- operationType String, corrispondente al campo del tracciato *tipo_operazione*, che contiene il codice di due caratteri numerici che indica il tipo operazione
- circuitType String, corrispondente al campo del tracciato *tipo_circuito*, che contiene il codice di due caratteri numerici che indicano il circuito di pagamento
- idTrxIssuer -> String, corrispondente al campo del tracciato *id_trx_issuer*, che contiene il Codice autorizzativo rilasciato dall' Issuer
- correlationId String, corrispondente al campo *correlation_id*, che contiene l' Identificativo di correlazione fra operazione di pagamento ed eventuale storno/reversal.
- amount Long, corrispondente al campo del tracciato *total_amount*, contenente l'importo della transazione valorizzata in centesimi di euro, come valore assoluto (segno corrispondente al codice operazione)
- amountCurrency String, corrispondente al campo *currency* del tracciato, stringa codificata con tre caratteri della codifica ISO (valore fisso 978)
- mcc String, corrispondete al campo *mcc* del tracciato. Il merchant category code attualmente è presente nel tracciato, ma verrà mascherato in fase di elaborazione dal processo batch con valore *00000*
- acquirerId String, corrispondente al campo del tracciato *acquirer_id*
- merchantId String, corrispondente al campo del tracciato *merchant_id*
- terminalId String
- bin String, campo che corrispondente al campo *bin* del tracciato, contenente un codice con le prime 8 cifre dello strumento di pagamento
- par String, campo corrispondente al campo *par* del tracciato.
- lineNumber Long, campo che indica la riga del tracciato contenente le informazioni della transazione in elaborazione
- filename String, campo che contiene il file da cui sono state estratte le informazioni della transazione in elaborazione

Il componente contiene inoltre le liste di hpan/token e par, impiegati per la fase di elaborazione delle transazione. Le informazioni relative alle transazioni, e le liste di HPAN e PAR, sono mantenute temporaneamente in memoria per permettere l'esecuzione del processo batch.

Il processo batch gestisce, come principale entità logica del processo di elaborazione, la rappresentazione delle transazioni ottenute dalla lettura del tracciato, *InboundTokenData*, che si compongono di:

- tokenPAN String, corrispondente al campo del tracciato *tokenPAN*
- circuitType String, corrispondente al codice identificativo del circuito della transazione da cui è estratto il token, utile a permettere l' inclusione di tutti i tracciati con circuito Mastercard
- par String, corrispondente al campo del tracciato *par*

Il componente contiene inoltre la lista di bin range, impiegati per la fase di elaborazione delle liste di token recuperate dagli acquirer. Le informazioni relative ai token, e la lista di Bin Range, sono mantenute temporaneamente in memoria per permettere l'esecuzione del processo batch.

Il componente contiene le entità derivate dal framework Spring Batch, che includono le informazioni sullo stato delle esecuzioni del processo batch.

Write access-patterns

I dati relativi alle liste di HPAN/TokenPAN e PAR, ottenute tramite i servizi esposti da CentroStella, sono scritti in blocchi di dimensione fissa, definita tramite configurazione applicativa. Similmente le transazioni scritte nel file di output sono trattate con scritture in blocco, limitato dal numero di record inizialmente letti, e filtrato dei tracciati che non hanno trovato corrispondenza con gli HPAN/Token o con i PAR.

In caso siano presenti scarti per eliminazione di singole transazioni, o nel caso la procedura sia configurata per non interrompersi in caso di singoli errori di validazione/scrittura (vedere paragrafo sulla Fault Tolerance), le singole entità saranno riportate in file di scarto, con scritture singole.

I dati relativi alla lista di Bin Range, ottenute tramite il servizio esposto da CentroStella, sono scritti in memoria, con blocchi di dimensione fissa, definita tramite configurazione applicativa. Similmente i record da inserire neli file di output sono trattate con scritture in blocco, limitato dal numero di record inizialmente letti, e filtrato dai TokenPAN per cui non è stata trovata un range in cui il TokenPAN è rientrato in lista di Bin Range (oppure sia un record con circuito Mastercard), oppure è stato individuato un match fra l'hash del tokenPAN e la lista di HTokenPAN fornita dal TKM.

In caso siano presenti scarti, o nel caso la procedura sia configurata per non interrompersi in caso di singoli errori di validazione/scrittura (vedere paragrafo sulla Fault Tolerance), le singole entità saranno riportate in file di scarto, con scritture singole. Il tracciato di output sarà scritto rimuovendo il campo relativo al circuit_type.

Il metodo di scrittura utilizzato, per limitare impatti in performance, è non bloccante sia per la produzione dei file regolari, che per quelli in cui sono riportati gli scarti dell'applicativo.

Il processo prevede un'aggiornamento delle informazioni sullo stato di elaborazione del batch nel repository del framework Spring Batch, che viene aggiornato con singole chiamate sulle entità corrispondenti allo stato di esecuzione generale, e per le singole fasi del processo. Lo stato dell'elaborazione di ogni file in input al processo batch è aggiornato al termine della scrittura di ogni singolo blocco estratto dai file.

Read access-patterns

Il componente elabora i tracciati contenenti le transazioni da filtrare, e le liste di HPAN e PAR, con delle letture in bulk, la cui dimensione è data da configurazione a livello applicativo.

Il componente elabora i tracciati contenenti i tokenPAN, provenienti dalle transazioni filtrate nella prima fase di elaborazione del batch, la lista di Bin Range, e la lista di HTokenPAN proveniente dal TKM, con delle letture in bulk, la cui dimensione è data da configurazione a livello applicativo.

Tramite le funzionalità offerte dal framework Spring Batch, viene definito un processo in cui parallelamente possono essere trattati più tracciati in contemporanea, e per ogni file possono essere elaborati più blocchi di record in parallelo. Il livello di parallelismo fra file e blocchi in lettura è definito da configurazione applicativa.

Data-store e modello di dominio

Il componente batch prevede l'utilizzo di strutture dati in-memory per le entità corrispondenti alle transazioni e liste di HPAN e PAR, originariamente presenti sui rispettivi file, che saranno mantenute temporaneamente in memoria per il tempo necessario all'elaborazione dei tracciati.

Dato l'alto volume di informazioni trattate, sia per il numero di strumenti attivi in CentroStella, che per l'introduzione delle informazioni legate alle carte tokenizzate, è previsto l'utilizzo di file temporanei su disco, dove saranno salvate le informazioni in stato *parzialmente elaborato*. In particolare le liste HPAN/PAR non saranno interamente introdotte in memoria, ed i flussi di transazioni da elaborare saranno quindi oggetto di un' elaborazione su più cicli in cui saranno prodotti:

- Un file di transazioni scartate nel ciclo di corrente, che sarà confrontato con i dati ancora non caricati in memoria. Il file temporaneo sarà rimosso al termine dell'elaborazione su tutti i blocchi di HPAN e PAR ancora non letti
- Un file di transazioni di output parziale, che sarà considerato parziale fino al termine dell'elaborazione.

Le transazioni errate per errori di validazione saranno riportate nel file di scarto per errore, mentre solo le transazioni che non hanno trovato riscontro in tutti i cicli di elaborazione saranno riportate nel file degli scarti permanente, al termine dell'elaborazione.

Il componente batch prevede l'utilizzo di strutture dati in-memory per le entità corrispondenti alle lista relativa ai TokenPAN estratti dai tracciati delle transazioni, la lista di Bin range, e la lista di HTokenPAN fornita dal TKM, originariamente presenti sui rispettivi file, che saranno mantenute temporaneamente in memoria per il tempo necessario all'elaborazione dei tracciati.

Le entità logiche corrispondenti allo stato dell'esecuzione, derivate dall'utilizzo del framework Spring Batch, sono trattate in un repository che può essere in-memory, oppure su altro DB a disposizione del soggetto acquirer. La struttura del modello delle entità del framework è riportata nella documentazione ufficiale, disponibile al link: https://docs.spring.io/spring-batch/docs/current/reference/html/schema-appendix.html

Monitoring

Il componente è impiegato sugli ambienti dei singoli Acquirer, il monitoraggio è demandato ai singoli soggetti. Il componente batch, tramite le informazioni di meta-data disponibili nelle entità del framework utilizzato, può essere monitorato sullo stato delle singole esecuzioni. I soggetti acquirer possono quindi verificare:

- Stato delle elaborazioni effettuate sul batch, e delle singole fasi di elaborazione
- Tempo di elaborazione delle elaborazioni del batch nel complessivo, e nelle singole fasi
- Record letti, filtrati e scritti nelle singole fasi di elaborazione

Modelli di *fault-tolerance*

Il componente prevede una serie di casi per cui l'elaborazione del batch può non essere interrotta nel caso di errori avvenuti per singoli record, nel corso del processo. il componente può quindi essere configurato per fare in modo che in casi di errori sui singoli tracciati di un file si possa proseguire, anche a fronte di errori in lettura, validazione o scrittura.

Il numero di casi d'errore tollerati per un flusso, prima di bloccare la procedura, è definito da una configurazione applicativa, e può essere impostata in modo da produrre sempre l'interruzione del processo in ogni caso d'errore incontrato.

Errori in fase di comunicazione sui servizi di CentroStella, nella criptazione del file di output, o nell'invio su canale SFTP producono obbligatoriamente un blocco della procedura.

Scalabilità

Il componente è strutturato per eseguire la procedura in modo che una singola istanza gestisca tutti i record dei flussi in input. La singola istanza può essere scalata variando le risorse allocate per il processo, e le configurazioni per l'elaborazione di file e chunk su più thread. Possono essere utilizzate multiple istanze del processo, su cui l'acquirer potrà ripartire i tracciati da elaborare, **configurando l'applicativo per avere identificativi diversi nel caso utilizzino lo stesso database come repository.**

Componente *1*

Responsabilità

Descrivere qual'è il perimetro di responsabilità di questa componente: che ruolo gioca nel contesto più largo, di che porzione di dominio dati è owner.

Dipendenze

Indicare quali sono le dipendenze up-stream (chi chiama questa componente) e down-stream (chi è chiamato da questa componente).

Data-layer

Entità logiche

Semplice insieme delle entità logiche di responsabilità del dominio.

Write access-patterns

Per ogni entità logica: come vengono scritti i dati, singolarmente o in bulk? Quali sono le dimensioni attese delle entità (95-percentile, 90-percentile, …, mediana)?

Read access-patterns

Per ogni entità logica: come vengono letti i dati, singolarmente o in bulk? Quali sono le proprietà di ricerca? Quali sono le dimensioni attese delle entità? Paginazione necessaria?

Data-store e modello di dominio

Il data-store scelto ed Il modello fisico con cui rappresentare le entità logiche. Qual'è la durabilità attesa? Quante repliche? Dove?

Monitoring

Come viene monitorato il layer? Quali sono le metriche? Quali sono gli allarmi? Quale è il periodo?

Modelli di *fault-tolerance*

Quali sono gli scenari di fallimento della componente? Cosa si può rompere? Come reagisce per mantenere continuità?

Scalabilità

Come si pensa di gestire la scalabilità? Verticale? Orizzontale? Scale-out (aumentare)? Scale-in (ridurre)?

API *1*

Interna	*Y/N*
Esterna	*Y/N*
Sincrona / Asincrona	*S/A*
Autorizzazione	*Quale, se presente, strumento di autorizzazione?*
Autenticazione	*Quale, se presente, strumento di autenticazione?*
TPS (stimato)	*Quante invocazioni/sec. sono attese?*
Idempotente	*Y/N*
Stateless	*Y/N* *Se N, motivare il perché.*
Read / Write intensive	*R/W*
Cacheable	*Y/N*

Signature

Descrizione della segnature della API.

Errori

Quali errori può restituire? Cosa indicano?

Request

Modello degli input della API.

Response

Modello dell'output della API.

Monitoring

Come viene monitorato il layer? Quali sono le metriche? Quali sono gli allarmi? Quale è il periodo?

API *n*

Vedi API 1.

Componente *n*

Vedi Componente 1.

Operational Excellence (OE)

SLAs

Descrivere le SLA esposte ai consumatori del servizio. Nel formularle, tieni conto delle SLA delle tue dipendenze: una catena è tanto robusta quanto il suo anello più debole.

Availability

Descrivere come viene misurata l'availability del prodotto/funzione. Non confondere availability con essere up & running.

L'availability è funzione di:

1. *intervallo di tempo di musirazione*
2. *metriche di input*
3. *SLA per metrica di input*
4. *SLA complessiva*

Ad es., per un sistema quale YouTube la “5 minutes availability” A(5), può essere definita come:

```
A(5) = 2XX_streaming / tot_streaming > 0.99 and 2XX_uploads / tot_uploads > 0.97
```

Dashboard

Sketch della dashboard con le metriche che ti aiuteranno a garantire la OE del sistema.