Ministerstwo Finansów

Centralne Repozytorium danych o sprzedaży detalicznej

Usługi dodatkowe dla środowiska testowego

Departament Informatyzacji

Wersja 1.07

Spis treści

Uwaga ogólna

Usługi wymienione w rozdziałach 1 – 7 wymagają jednostronnego protokołu TLS z uwierzytelnieniem serwera.

1. Usługa – statusy komendy

Usługa umożliwia uzyskanie informacji na temat wysłanych komend. Jest to szczególnie przydatne w sytuacji, gdy wysłane dane nie zostały zapisane w repozytorium z powodu błędów, które wystąpiły na wcześniejszych etapach. Dane sprzed momentu „wyzerowania" historii fiskalizacji kasy nie są udostępniane. Dane, które nie zostały jeszcze pobrane z chmury Azure również nie zostaną udostępnione.
W przypadku zapytania o komendę typu DFD (wysyłka danych) usługa zwróci informacje dotyczące: statusów przetwarzania, błędów, szczegółów przetwarzania pliku. Dla pozostałych komend usługa zwróci informacje na temat: statusu przetwarzania, treści komendy, czasu wykonania, autora wysyłki, kodu i komunikatu z Web API.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/command/{commandId}/statuses**
Nazwa	**Opis**
-commandId	Kod komendy w formacie „XXX.AAAXXXXXXXXXX.date-time" gdzie: • XXX – identyfikator komendy i składa się z 3 znaków. • AAAXXXXXXXXXX - numer unikatowy urządzenia fiskalnego. • date-time - YYYY-MM-DDTHH:MM:SS.SSSZ. np. „DFD.AAA1234567890.2017-12-07T11:08:39.351Z"

Odpowiedź w przypadku zapytania o komendę typu DFD:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-statuses	Lista statusów
--status	Status/etap przetwarzania pliku
--code	Ewentualny kod błędu
--correlationId	Numer korelacyjny, czyli identyfikator powiązanej komendy, umieszczany w nagłówku JWS
--timestamp	Data i czas uzyskania statusu
-jwsObject	Zaszyfrowany plik z danymi
--payload	Base64 z zaszyfrowanymi danymi
--timestamp	Data i czas zapisu pliku
-jpkObject	Rozszyfrowane dane JPK
--payload	Plik JPK
--timestamp	Data i czas zapisu pliku
-errObject	Informacje o błędzie
--payload	JSON z ewentualnymi szczegółami o błędzie
--timestamp	Data i czas zapisu informacji

Odpowiedź w przypadku zapytania o komendy innego typu niż DFD:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-command	Informacje dotyczące komendy
--status	Status komendy
--content	Treść komendy
--timestamp	Data i czas wykonania komendy pobranej z Web API

--sentBy	Autor komendy: ‘K’ – kasa, ‘S’ - system
--codeWebApi	Status wykonania komendy pobranej z Web API
--messageWebApi	Opis błędu w przypadku problemów z wykonaniem komendy

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/command/DFD.ZTE1701000901.2018-01-22T06:07:41.168Z/statuses

Przykład odpowiedzi:

```
{"statuses":[
   {"status":"validated","code":"0",
   "crrelationId":"…","timestamp":"2017-10-29T20:51:18.660Z"},
   {"status":"verified","code":"0",
   "crrelationId":"…",  "timestamp":"2017-10-29T20:51:18.634Z"},
   {"status":"received","code":"0",
   "crrelationId":"…",  "timestamp":"2017-10-29T20:50:42.910Z"}
],
"jwsObject":{"payload":"eyJqcGt…","timestamp":"2017-10-28T17:27:09.794Z"},
"jpkObject":{
   "payload":{"JPK":{"…"}}},"timestamp":"2017-10-28T17:27:09.929Z"},
"errObject":{"payload":"","timestamp":""}}
```

Obiekt errObject zawiera informacje o błędach, które wystąpiły do czasu utrwalenia danych w repozytorium. Np. przy weryfikacji podpisu i szyfrowania mogą wystąpić następujące błędy:

```
*RSA signature did not verify*                  niepoprawny certyfikat podpisujący;
*RSA decryption failed*                         niepoprawny certyfikat szyfrujący;
Signer certificate validation failed. Certificate not trusted    nieznany certyfikat
podpisujący.
```

Komenda może przyjmować następujące statusy:

```
200 - received       odebrano poprawną paczkę komendy;
202 - verified       paczka poprawnie zaszyfrowana i podpisana;
203 - validated      dokument JPK o poprawnej strukturze zgodnej ze schematem;
401 - unprocessed    niepoprawne metadane paczki, niepoprawne kompresowanie danych;
402 - unverified     paczka niepoprawnie zaszyfrowana lub niepoprawnie podpisana;
403 - unvalidated    dokument JPK o niepoprawnej strukturze lub niezgodny ze schematem.
```

W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```

gdzie httpStatus przyjmuje następujące wartości:

401 – nieprawidłowy format parametru commandid,

300 – komenda nie została utrwalona w repozytorium bądź wykonano później usługę „zerowania” historii fiskalizacji kasy,

301 – błąd odczytu z bazy danych,

302 – nieznany inny błąd.

2. Usługa – historia komend kasy

Usługa umożliwia uzyskanie informacji o identyfikatorach komend zgromadzonych w repozytorium dla wskazanej kasy(które zostały wysłane do kasy lub przekazane przez kasę). Dane sprzed momentu „wyzerowania" historii fiskalizacji kasy nie są udostępniane.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/cashRegister/{uniqueNr}/commands**
Nazwa	**Opis**
-uniqueNr	Numer unikatowy urządzenia fiskalnego

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-commands	Komendy dotyczące odpytywanej kasy
--sent	Lista kodów komend (commandId), które zostały wysłane do kasy przez system
--received	Lista kodów komend (commandId), które zostały wysłane przez kasę

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/cashRegister/ZTE1701000901/commands

Przykład odpowiedzi:

```
{"commands":{"sent":["TFD.ZTE1701000901.2018-01-23T13:53:31.249Z"
],
"received":[
"DFD.ZTE1701000901.2017-10-19T07:16:00.169Z",
"DFD.ZTE1701000901.2017-10-19T11:21:51.130Z",
"DFD.ZTE1701000901.2017-10-19T11:38:45.950Z",
"DFD.ZTE1701000901.2018-01-10T13:23:54.587Z"]}
}
```

W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```

gdzie httpStatus przyjmuje następujące wartości:
401 – nieprawidłowy format parametru uniqueNr,
300 – kasa nie została utrwalona w repozytorium bądź wykonano później usługę „zerowania" historii fiskalizacji kasy,
301 – błąd odczytu z bazy danych,
302 – nieznany inny błąd.

3. Usługa – raport z fiskalizacji

Usługa umożliwia uzyskanie dokumentu z raportem fiskalizacji dla wskazanej kasy oraz informacje o podatniku wraz z adresem punktu sprzedaży. Dane sprzed momentu „wyzerowania" historii fiskalizacji kasy nie są udostępniane.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/cashRegister/{uniqueNr}/report/fiscal**
Nazwa	**Opis**
-uniqueNr	Numer unikatowy urządzenia fiskalnego

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-documents	Dokumenty z raportem z fiskalizacji, dotyczące odpytywanej kasy
--payload	Raport z fiskalizacji lub informacje o podatniku
--timestamp	Data i czas zapisu dokumentu w bazie repozytorium

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/cashRegister/ZTE1701000901/report/fiscal

Przykład odpowiedzi:


```
{"documents":[
{"payload":{"rapFisk":{"serwis":"a","zakRap":"2017-02-
14T16:50:03.342Z","firmwareId":"a","typWlasn":"1","kodUS":"0000","serwID":"a","pamiecCh
r":1,"sposobUzytk":"1","serwName":"a","licznikParag":1,"JPKID":1,"dataFisk":"2017-02-
14T16:50:03.342Z","kasjer":"a","stPTU":[{"id":"A","wart":2300},{"id":"B","wart":800},{"
id":"C","wart":500},{"id":"D","wart":0},{"id":"E","wart":1},{"id":"F","wart":10000},{"i
d":"G","wart":"ZW"}],"model":"a","waluta":"AAA","katKasy":["00"],"nrDok":1,"nrKasy":"a"
,"podpis":{"RSA":"01FF"}}},
"timestamp":"2018-01-22T06:46:57.718Z"},
{"payload":{"rapFisk":{"serwis":"a","zakRap":"2017-02-
14T16:50:03.342Z","firmwareId":"a","typWlasn":"1","kodUS":"0000","serwID":"a","pamiecCh
r":1,"sposobUzytk":"1","serwName":"a","licznikParag":1,"JPKID":11,"dataFisk":"2017-02-
14T16:50:03.342Z","kasjer":"a","stPTU":[{"id":"A","wart":"ZW"},{"id":"A","wart":"ZW"},{
"id":"A","wart":"ZW"}],"model":"a","waluta":"AAA","katKasy":["01","02","03","04","05","
06"],"nrDok":1,"nrKasy":"a","podpis":{"RSA":"0BFF","SHA":"0BFE"}}},
"timestamp":"2018-01-22T06:46:57.718Z"}]}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```

gdzie httpStatus przyjmuje następujące wartości:
401 – nieprawidłowy format parametru uniqueNr,
300 – kasa nie została utrwalona w repozytorium bądź wykonano później usługę „zerowania" historii fiskalizacji kasy,
301 – błąd odczytu z bazy danych,
302 – nieznany inny błąd.

4. Usługa – dokument kasy

Usługa umożliwia uzyskanie zawartości dokumentu zapisanego w repozytorium dla wskazanej kasy. Dane sprzed momentu „wyzerowania" historii fiskalizacji kasy nie są udostępniane. W przypadku wysłania do repozytorium kilku dokumentów z tymi samymi numerami jpkId oraz memNr, zostanie zwrócony ostatni odebrany taki dokument.

Uwaga! Jeśli pomimo wysyłania danych do repozytorium, otrzymywany jest kod błędu 300, sygnalizujący brak utrwalenia danych w repozytorium, warto sprawdzić, czy nie wystąpiły błędy na wcześniejszym etapie. W tym celu należy skorzystać z usługi „historia komend kasy", aby odczytać identyfikatory poszczególnych komend DFD. Następnie, dla każdej takiej komendy można wykonać usługę „statusy komendy", aby sprawdzić, czy wystąpiły błędy podczas przetwarzania przesyłki przez system.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/cashRegister/{uniqueNr}/document/{memNr}/{jpkId}**
Nazwa	**Opis**
-uniqueNr	Numer unikatowy urządzenia fiskalnego
-memNr	Numer pamięci chronionej
-jpkId	Identyfikator dokumentu JPK

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-documents	Dokumenty, dotyczące odpytywanej kasy
--payload	Pojedynczy dokument
--timestamp	Data i czas zapisu dokumentu w bazie repozytorium

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/cashRegister/ZTE1701000901/document/1/2

Przykład odpowiedzi:

```
{"documents":[
{"payload":{"paragon":…},
"timestamp":"2018-01-15T21:03:03.760Z"}]
}
```

W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```

gdzie httpStatus przyjmuje następujące wartości:
401 – nieprawidłowy format parametru uniqueNr,
405 – nieprawidłowy format parametru memNr,
407 – nieprawidłowy format parametru jpkId,
300 – kasa nie została utrwalona w repozytorium bądź wykonano później usługę „zerowania" historii fiskalizacji kasy,
301 – błąd odczytu z bazy danych,
302 – nieznany inny błąd.

5. Usługa – dane kasy (raport nr 1)

Usługa umożliwia uzyskanie informacji o danej kasie fiskalnej. Dane sprzed momentu „wyzerowania" historii fiskalizacji kasy nie są udostępniane. Usługa jest szczególnie użyteczna, jeśli chcemy poznać, jaki jest obecnie status kasy. Możliwe zwracane przez usługę statusy kasy:

POL – kasa połączona do repozytorium,
ROZ – rozpoczęta fiskalizacja kasy,
FIS – zakończona fiskalizacja kasy,
WYR – kasa wyrejestrowana,
RDO – kasa w trybie „tylko do odczytu".

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/cashRegister/{uniqueNr}**
Nazwa	**Opis**
-uniqueNr	Numer unikatowy urządzenia fiskalnego

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-cashRegister	Informacje o kasie fiskalnej
--uniqueNumber	Numer unikatowy urządzenia fiskalnego
--updatedDate	Data i czas modyfikacji informacji o urządzeniu w bazie repozytorium
--cashRegisterModel	Informacje o modelu urządzenia
---modelName	Nazwa modelu
---producerTin	NIP producenta
---updatedDate	Data i czas modyfikacji informacji o modelu urządzenia w bazie repozytorium
--recordNumber	Numer ewidencyjny urządzenia
--publicKeyCertificate	Informacje o certyfikacie urządzenia
---serialNumber	Numer seryjny certyfikatu
---validFromDate	Data, od kiedy ważny jest certyfikat
--factoryIdNumber	Numer fabryczny urządzenia
--status	Status kasy
--fiscalizationDate	Data fiskalizacji kasy
--taxOffice	Kod oraz nazwa urzędu skarbowego
--serviceman	Informacje o serwisancie
---servicemanId	ID serwisanta
---updatedDate	Data modyfikacji informacji o serwisancie
--taxpayerTin	NIP podatnika
--deregistrationDate	Data wyrejestrowania
--lastDocSentDate	Data i czas przysłania ostatniego dokumentu przez kasę
--docsNumber	Liczba przesłanych dokumentów do momentu wyrejestrowania

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/cashRegister/ZTE1701000901

Przykład odpowiedzi:


```
{"cashRegister":
        {"uniqueNumber":"ZAC1701001251",
```

```
        "updatedDate":"2018-01-30T09:16:08.580Z",
        "cashRegisterModel":
                {"modelName":"Urządzenie 123 test",
                "producerTin":"9999999999",
                "updatedDate":"2017-12-18T10:17:13.924Z"},
        "recordNumber":"2018/000000025",
        "publicKeyCertificate":
                {"serialNumber":"642840192000000000",
                "validFromDate":"2018-01-29T23:00:00.000Z"},
        "factoryIdNumber":"TTTT12345726",
        "status":"FIS",
        "fiscalizationDate":"2018-01-29T23:00:00.000Z",
        "taxOffice":"1434 URZĄD SKARBOWY WARSZAWA-PRAGA",
        "serviceman":{"updatedDate":""},
        "taxpayerTin":"5220001694",
        "deregistrationDate":"",
        "lastDocSentDate":"2018-02-09T20:55:42.496Z",
        "docsNumber":"38"}
}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:


```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```


gdzie httpStatus przyjmuje następujące wartości:

401 – nieprawidłowy format parametru uniqueNr,

300 – kasa nie została utrwalona w repozytorium bądź wykonano później usługę „zerowania” historii fiskalizacji kasy,

301 – błąd odczytu z bazy danych,

302 – nieznany inny błąd.

6. Usługa – dane producenta kasy fiskalnej (raport nr 2)

Usługa umożliwia uzyskanie informacji o producencie kasy fiskalnej.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/producer/{tin}**
Nazwa	**Opis**
-NIP	NIP producenta

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-fullName	Nazwa producenta
-tin	NIP producenta
-updatedDate	Data i czas modyfikacji informacji o producencie
-updatedBy	Informacja o operatorze, który dokonał zmiany w bazie repozytorium
-phoneNumber	Numer telefonu producenta
-city	Miejscowość w danych adresowych producenta
-houseNumber	Numer domu
-street	Nazwa ulicy
-flatNumber	Numer domu
-postalCode	Kod pocztowy
-postOffice	Poczta

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/producer/5220001694

Przykład odpowiedzi:

```
{"producer":
        {"fullName":"TEST FISKALIZACJI",
        "tin":"5220001694",
        "updatedDate":"2017-12-20T08:38:56.645Z",
        "updatedBy":"ATEK",
        "city":"WARSZAWA",
        "houseNumber":"20",
        "street":"KASIASTA",
        "postOffice":"WARSZAWA"}
}
```

W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```

gdzie httpStatus przyjmuje następujące wartości:
401 – nieprawidłowy format parametru tin,
300 – brak danych o producencie w repozytorium,
301 – błąd odczytu z bazy danych,
302 – nieznany inny błąd,
303 – brak modeli kas dla wskazanego producenta.

7. Usługa – dane o homologacjach modelu urządzenia fiskalnego (raport nr 3)

Usługa umożliwia uzyskanie informacji o homologacjach modelu kasy fiskalnej.

Wywołanie:

Wywołanie	**GET /crr-ext-rest-services/public/cashRegisterModel/{uniqueNrPrefix}/homologations**
lub wywołanie	**GET /crr-ext-rest-services/public/cashRegisterModel/{uniqueNrPrefix}/homologations/all**
Nazwa	**Opis**
-uniqueNrPrefix	Prefiks numeru unikatowego urządzenia fiskalnego

Odpowiedź:

Odpowiedź	**http 200 OK**
Nazwa	**Opis**
-modelName	Nazwa modelu
--homologations	Lista informacji o ważnych lub wszystkich homologacjach
---firmwareMetadata	Wersja oprogramowania
---firmwareChecksum	Suma kontrolna
---updatedDate	Data i czas modyfikacji informacji o homologacji w bazie repozytorium
---updatedBy	Informacja o operatorze, który dokonał modyfikacji w bazie repozytorium
--validFromDate	Data, od której ważna jest homologacja
--validToDate	Data, do kiedy ważna jest homologacja

Przykład wywołania:

https://test-crr.mf.gov.pl/crr-ext-rest-services/public/cashRegisterModel/ARR/homologations/all

Przykład odpowiedzi:


```
{"modelName":"Arek test test3",
"homologations":
        [{"firmwareMetadata":"Test 3 12.0",
        "firmwareChecksum":"WifHworix6sjdieckjdnhwleifussldo4sldkfrj349f234",
        "updatedDate":"2017-12-12T10:22:46.859Z",
        "updatedBy":"ATEK"},
        {"firmwareMetadata":"Linux 12.0.3 test2",
        "firmwareChecksum":"WifHworix6sjdieckjdnhwleifussldo4sldkfrj349f987",
        "updatedDate":"2017-12-12T10:21:13.597Z",
        "updatedBy":"ATEK",
        "validFromDate":"2018-01-01",
        "validToDate":"2018-12-31"}]
}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:


```
{"crrError":
        {"crrMessage":"Opis błędu",
        "httpStatus":XXX}
}
```


gdzie httpStatus przyjmuje następujące wartości:
401 – nieprawidłowy format parametru uniqueNrPrefix,
300 – brak danych o modelu kasy w repozytorium,

301 – błąd odczytu z bazy danych,
302 – nieznany inny błąd.

8. Samodzielne generowanie komend dla kasy

W środowisku testowym istnieje funkcjonalność pozwalająca na samodzielne generowanie komend wysyłanych do kas.

Ze stanowiska klienta chcącego przeprowadzić operację generowania komend, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie operacji generowania komend tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc generowanie komend dla kasy może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co kasa dla której mają być wygenerowane komendy.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:


```
{
  "code": "B10",
  "message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",
  "requestId": "Nieodp wyst cert"
}
```


W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": ""
}
```


Możliwe jest samodzielne generowanie komend wysyłanych do kas dla następujących komend :

CCS	CMD01. Połącz z Serwerem CPD i wywołaj usługę	Call CPD Server
TFD	CMD02. Pobierz harmonogram transmisji danych	Timetable For Device
CQC	CMD03. Włącz/wyłącz drukowanie kodu QR	Control QR Code
DDD	CMD06. Pobierz szczegółowe dane z urządzenia fiskalnego	Detailed Device Documents

- Składnia żądania do generowania komendy CMD01 Połącz z serwerem CPD i wywołaj usługę.

Żądanie z elementem "cpdServiceName": "XXX” pozwoli wygenerować dowolną komendę (brak walidacji treści pola cpdServiceName),
żądanie bez elementu "cpdServiceName" generuje komendę z "cpdServiceName" : "KFD”

Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZAX170100XXXX.2017-08-14T16:50:03.342Z",
   "command": {
    "attributes": {
          "typ":"CCS",
           "cpdServiceName": "KCS"

   }
}
}
```


- Składnia żądania do generowania komendy CMD02 Pobierz harmonogram transmisji danych.

Treść żądania z elementami „sendFreqEventHub", „checkFreqWebApi", „shippmentType” pozwoli wygenerować komendę z określonymi wartościami pól (brak walidacji treści pól),

Jeśli chodzi o elementy „sendFreqEventHub, checkFreqWebApi, shippmentType" – to można również wysłać żądanie tylko z jednym z nich lub dwoma lub trzema w zależności, który element chcemy zindywidualizować (pozostałe elementy wygenerują się z wartościami standardowymi (opis poniżej).

Żądanie bez elementów generuje komendę „TFD” ze standardowymi wartościami (sendFreqEventHub=7200,checkFreqWebApi= 1800; shippmentType="PA;FA;RD;WN").

Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZAX170100XXXX.2017-08-14T16:50:03.342Z",
  "command": {
    "attributes": {
```

```
            "typ":"TFD",
        "sendFreqEventHub": 7777,
        "checkFreqWebApi": 600,
        "shippmentType": "PA"
                    }
            }
}
```


- Składnia żądania do generowania komendy CMD03. Włącz/wyłącz drukowanie kodu QR.

Treść żądania z dodatkowym elementem "qrCode" pozwoli wygenerować komendę z określoną wartością pola qrCode (brak walidacji treści pola), żądanie bez elementu "qrCode" generuje komendę „CQC” ze standardową wartością pola qrCode=1000

Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
 "commandId": "TTT.ZAX170100XXXX.2017-08-14T16:50:03.342Z",
   "command": {
     "attributes": {
           "typ":"CQC",
            "qrCode": 444
                    }
           }
}
```


- Składnia żądania do generowania komendy CMD06. Pobierz szczegółowe dane z urządzenia fiskalnego.

Treść żądania z elementami "scope", "shippmentType": pozwoli wygenerować komendę z określoną wartością pól scope oraz shipmentType (brak walidacji treści pól); tu można również wykonać requesta tylko z jednym z nich lub dwoma w zależności, który element chcemy zindywidualizować (pozostały element wygeneruje się z wartościami standardowym.i

Żądanie bez tych elementów generuje komendę „DDD” ze standardową wartością pól - scope="1000-2000", shippmentType="RF;PA;FA;RD;WN"

Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command

```
{
"commandId": "TTT.ZAX170100XXXX.2017-08-14T16:50:03.342Z",
  "command": {
    "attributes": {
          "typ":"DDD",
           "scope": "123-234",
           "shippmentType": "PA"
                        }
          }
}
```

9. „Zerowanie” historii fiskalizacji kasy

Możliwe jest „zerowanie” stanu historii fiskalizacji kasy – tj. przywrócenie zapisów w Repozytorium obejmujące zapisy związane z procesem fiskalizacji kasy do stanu sprzed tego procesu.

W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": ""
}
```


Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZTE1701000901.2017-11-05T16:50:03.342Z",
  "command": {
    "attributes": {
"typ":"DEL"
}
}
}
```


Ze stanowiska klienta chcącego przeprowadzić operację „zerowania”, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie operacji „zerowania” tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc „zerowania” kasy może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co „zerowana” kasa.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:


```
{
  "code": "B10",
  "message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",
```

```
  "requestId": "Nieodp wyst cert"
}
```


10. Ustawienie w Repozytorium CPD wartości firmwareMetadata oraz firmwareChecksum.

Za pomocą poniższego komunikatu możliwe jest ustawienie w repozytorium aktualnej wartości firmwareMetadata oraz firmwareChecksum – do testowania odpowiedzi komendy CMD10.

Uwaga: użycie komendy zamyka wszystkie istniejące w repozytorium wpisy z firmware i wstawia jeden ważny (wprowadzany tym poleceniem) firmware.

W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": ""
}
```


Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZTE1701000901.2017-11-05T16:50:03.342Z",
  "command": {
    "attributes": {
            "typ":"CNF",
      "firmwareMetadata": "ver 12.22",
      "firmwareChecksum": "68e656b251e67e8358bef8483ab0d51c6619f3e7a1a9f0e75838d41ff368f728"

}
}
}
```

Ze stanowiska klienta chcącego przeprowadzić operację ustawienia w Repozytorium CPD wartości firmwareMetadata oraz firmwareChecksum, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie tej operacji tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc operację tą może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co aktualizowana kasa.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:


```
{
  "code": "B10",
  "message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",
  "requestId": "Nieodp wyst cert"
}
```


11. Dodanie w Repozytorium CPD nowego firmwareMetadata oraz firmwareChecksum.

Za pomocą poniższego komunikatu TTT możliwe jest dodanie w repozytorium nowego wpisu dotyczącego wartości firmwareMetadata oraz firmwareChecksum. Usługa sprawdza czy istnieje już w repozytorium ważny rekord o podanym firmwareMetadata i w takim przypadku nie dodaje takiego samego firmwareMetadata tylko zwraca komunikat odpowiedzi HTTP/1.1 400 Bad Request (poprawne dodanie rekordu daje odpowiedź HTTP/1.1 204 No Content).

W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": ""
}
```


Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command

```
{
"commandId": "TTT.ZTE1701000901.2017-11-05T16:50:03.342Z",
  "command": {
    "attributes": {
            "typ":"CNF_INS",
     "firmwareMetadata": "ver 12.40",
     "firmwareChecksum": "68e656b251e67e8358bef8483ab0d51c6619f3e7a1a9f0e75838d41ff368f728"

}
}
}
```


Ze stanowiska klienta chcącego przeprowadzić operację dodawania w Repozytorium CPD nowego wpisu firmwareMetadata oraz firmwareChecksum, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie tej operacji tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc operację tą może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co aktualizowana kasa.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:


```
{
  "code": "B10",
  "message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",
  "requestId": "Nieodp wyst cert"
}
```


12. Zamknięcie ważności istniejącego w Repozytorium CPD wpisu z firmwareMetadata oraz firmwareChecksum.

Za pomocą poniższego komunikatu TTT możliwe jest zamknięcie ważności istniejącego w Repozytorium CPD wpisu dotyczącego zestawu wartości firmwareMetadata oraz firmwareChecksum. Usługa zamyka ważność dla istniejącego wpisu dla danego modelu kasy oraz dla podanego w komunikacie firmwareMetadata.

W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": ""
}
```


Przykład wywołania:

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZTE1701000901.2017-11-05T16:50:03.342Z",
  "command": {
    "attributes": {
            "typ":"CNF_CLS",
     "firmwareMetadata": "ver 12.40"

}
}
}
```


Ze stanowiska klienta chcącego przeprowadzić operację zamknięcia ważności w Repozytorium CPD określonego firmwareMetadata oraz firmwareChecksum, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie tej operacji tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc operację tą może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co aktualizowana kasa.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:


```
{
  "code": "B10",
  "message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",
  "requestId": "Nieodp wyst cert"
}
```

13. Ustawienie przez użytkownika komunikatu błędu dla wywoływanych komend.

W trakcie testów procesu fiskalizacji istnieje możliwość testowania kas pod kątem obsługi błędów zwracanych przez Repozytorium po wykonaniu komend.

Ze stanowiska klienta chcącego przeprowadzić operację ustawienia kodu błędu dla określonej komend, używającego do połączenia TLS do Serwera CPD certyfikatu, którego wystawcą jest określony producent, możliwe jest przeprowadzenie tej operacji tylko dla tych kas, których wystawca certyfikatu przesłany podczas jej procesu fiskalizacji komendą CMD08 jest taki sam. Inaczej mówiąc operację tą może dokonać jedynie klient posługujący się certyfikatem wystawionym przez tego samego wystawcę certyfikatu co aktualizowana kasa.

W przypadku niespełnienia powyższego warunku klient otrzyma odpowiedź:

{

"code": "B10",

"message": "Operacja zakonczona niepowodzeniem. Bledny certyfikat.",

"requestId": "Nieodp wyst cert"

}

Możliwe jest ustawienie kodów błędów w odpowiedzi na wywołanie komendy dla poniższych komend:

Identyfikator komendy	Nazwa komendy
KCS	CMD07. Pobierz klucze Repozytorium
KFD	CMD08. Wyślij certyfikaty urządzenia fiskalnego
MFP	CMD09. Wykonaj fiskalizację
CNF	CMD10. Sprawdź homologację nowego programu pracy urządzenia fiskalnego
TCS	CMD11. Test komunikacji z Serwerem CPD
GIS	CMD12. Pobierz ostatni stan Repozytorium
FPD	CMD14. Potwierdzenie poprawnej fiskalizacji
EFL	CMD15. Przejście kasy w tryb tylko do odczytu

Lista możliwych kodów błędu jest opisana w dokumentacji „Opis techniczny protokołu komunikacyjnego kasa – Centralne Repozytorium Kas – Specyfikacja komend” w tabeli „Możliwe kody błędów w odpowiedzi na wywołanie komendy:” dla każdej z komend.

Przykład żądania do ustawienia kodu błędu dla odpowiedzi na wykonanie komendy (w przykładzie komenda KCS i zwracany kod błędu E04) :

https://esb-te.mf.gov.pl:5062/api/SerwerCPD/Command


```
{
"commandId": "TTT.ZTE1701000901.2020-11-30T10:50:03.342Z",
  "command": {
    "attributes": {
      "typ":"SETERR",
            "CPDServiceName":"KCS",
            "code":"E04"
}
}
}
```


"typ" – typ komunikatu – stała wartość "SETERR" – ustawienie generowania komunikatu błędu lub "CLSERR" – anulowanie ustawienia generowania komunikatu błędu

"CPDServiceName" – identyfikator komendy

"code" – kod błędu

Przy wywołaniu przykładowej komendy dla kasy której ustawiono generowanie błędu pojawi się odpowiedź:


```
{
  "code": "E04",
  "message": "Wywolanie SQL zakonczone niepowodzeniem.",
  "requestId": "SymERR"
}
```


W przypadku kiedy GUM w trakcie testów procesu homologacji zablokuje kasę – operacja będzie niedostępna – w takim przypadku klient otrzyma odpowiedź:


```
{
```

```
  "code": "E02",
  "message": "Brak uprawnien do operacji - kasa zablokowana przez GUM.",
  "requestId": "GenErr"
}
```


Podanie w komunikacie wartości spoza wartości opisanych w tabeli spowoduje wygenerowanie przykładowej odpowiedzi:


```
{
  "code": "E02",
  "message": "Symulacja wystapienia bledu zakonczona niepowodzeniem, TYP:SETERR, CPDServiceName:FPD, CODE:E014",
  "requestId": ""
}
```


14. Operacje na łańcuchach dokumentów

Usługa wymaga jednostronnego protokołu TLS z uwierzytelnieniem serwera. Wszystkie operacje zwracają maksymalnie 20 dokumentów podczas jednego wywołania. Wszystkie parametry wywołania są wymagane. Usługa dla kas w postaci oprogramowania.

Dostępne operacje:

1. Listowanie dokumentów przesłanych do repozytorium
2. Weryfikacja ciągłości łańcucha paragonów
3. Weryfikacja ciągłości łańcucha paragonów anulowanych
4. Weryfikacja ciągłości łańcucha raportów dobowych

13.1 Operacja listowania dokumentów

Wywołanie:

Wywołanie	**GET /api/SerwerCPD/DocumentChain**
Nazwa	**Opis**

parametru	
- idKasy	Numer unikatowy urządzenia fiskalnego
- startJPKID	Numer początkowy JPKID, od którego będą pobierane kolejne dokumenty w porządku malejącym. Dla operacji listowania możliwe jest użycie `startJPKID=0010000000000000000`, które oznacza rozpoczęcie listowania od dokumentu o największym dostępnym JPKID dla danej kasy
- docType	Typ dokumentu, jednocześnie określenie rodzaju operacji. Dla operacji listowania `docType=list`

Odpowiedź:
Poprawna odpowiedź (HTTP/1.1 200 OK Content-Type: application/json) zawiera listę w porządku malejącym (wg JPKID) w układzie „JPKID”:”typ dokumentu”, ograniczoną do maksymalnie 20 pozycji.

Przykład wywołania:
https://esb-te.mf.gov.pl:5067/api/SerwerCPD/DocumentChain?idKasy=WTE2001000008&startJPKID=0010000000000000000&docType=list

Przykład odpowiedzi:

```
{
   "00100000000000031": "paragon",
   "00100000000000030": "paragAnul",
   "00100000000000029": "rapDob",
   "00100000000000028": "paragon",
   "00100000000000027": "paragon",
   "00100000000000026": "zdarzenie",
   "00100000000000025": "paragon",
   "00100000000000024": "paragAnul",
   "00100000000000023": "paragon",
   "00100000000000022": "paragon",
   "00100000000000021": "paragon",
   "00100000000000020": "paragAnul",
   "00100000000000019": "paragAnul",
   "00100000000000018": "paragon",
   "00100000000000017": "rapDob",
   "00100000000000016": "info",
   "00100000000000015": "paragAnul",
   "00100000000000014": "info",
   "00100000000000013": "paragAnul",
   "00100000000000012": "paragon"
}
```

W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON:

```
{
   "code": "B13",
   "message": "opis błędu"
}
```

Gdzie: `opis błędu` – string zawierający szczegóły błędu

13.2 Operacja weryfikacji ciągłości łańcucha paragonów

Wywołanie:

Wywołanie	GET /api/SerwerCPD/DocumentChain
Nazwa parametru	**Opis**
- idKasy	Numer unikatowy urządzenia fiskalnego

- startJPKID	Numer początkowy JPKID, od którego będą pobierane kolejne dokumenty w porządku malejącym.
- docType	Typ dokumentu, jednocześnie określenie rodzaju operacji. Dla operacji weryfikacji ciągłości łańcucha paragonów `docType=paragon`

Odpowiedź:

Poprawna odpowiedź (HTTP/1.1 200 OK Content-Type: application/json) zawiera listę w porządku malejącym (wg JPKID), ograniczoną do maksymalnie 20 pozycji w układzie:

```
"JPKID":{
   "SHA256": wartość
   "JPKREF":{
         "SHA256": wartość,
         "JPKID": wartość
    },
    "verification": wynik weryfikacji
}
```

gdzie:

`JPKID` – identyfikator dokumentu, element zawierający:

- `SHA256` – wartość funkcji skrótu dokumentu
- `JPKREF` – element zawierający:
  - `SHA256` - wartość funkcji skrótu poprzedniego dokumentu,
  - `JPKID` - identyfikator poprzedniego dokumentu,
- `verification` – wynik weryfikacji dokumentu. Pierwszy dokument na liście otrzymuje wynik weryfikacji „Start", ze względu na konstrukcję łańcucha paragonów. Następne jeśli poprawny, to „OK" w przeciwnym wypadku „Error" i dalsze sprawdzanie jest przerywane.

Dokładny opis tworzenia poszczególnych pól dostępny jest w dokumencie „Opis techniczny protokołu komunikacyjnego kasa – Centralne Repozytorium Kas – Standardy kryptograficzne" w rozdziale „A.10.3 Podpisywanie dokumentów w postaci elektronicznej".

Przykład wywołania:

https://esb-te.mf.gov.pl:5067/api/SerwerCPD/DocumentChain?idKasy=WTE2001000008&startJPKID=001000000000000031&docType=paragon

Przykład odpowiedzi:

```
{
   "00100000000000031":    {
      "SHA256": "84de47d2e43517829f9805dc1938bd94b2aee4049b09f42a10b83ec2e9893e17",
      "JPKREF":       {
         "SHA256":
"eb64237cd28302fca3786f6f218f4ad5fd2d00016c4270ee829539ea43603ea1",
         "JPKID": "00100000000000028"
      },
      "verification": "Start"
   },
   "00100000000000028":    {
      "SHA256": "eb64237cd28302fca3786f6f218f4ad5fd2d00016c4270ee829539ea43603ea1",
      "JPKREF":       {
         "SHA256":
"f44b4e24b1950153d4bcced4282f6d995c502835a46d3a3d7f7930065797921d",
         "JPKID": "00100000000000027"
      },
      "verification": "OK"
   },
```

```
   "001000000000000027":    {
      "SHA256": "f44b4e24b1950153d4bcced4282f6d995c502835a46d3a3d7f7930065797921d",
      "JPKREF":       {
         "SHA256":
"850fb10b026c349e733609573d02a65f81c282a72d810ad4f08581b8ac6e5ac4",
         "JPKID": "001000000000000025"
      },
      "verification": "OK"
   },
   "001000000000000025":    {
      "SHA256": "850fb10b026c349e733609573d02a65f81c282a72d810ad4f08581b8ac6e5ac4",
      "JPKREF":       {
         "SHA256":
"fb167c0cd9aed0ec6587c8f4a4b3bdc5847355307600d47e4f0cb56c39fed00a",
         "JPKID": "001000000000000023"
      },
      "verification": "OK"
   },
   "001000000000000023":    {
      "SHA256": "fb167c0cd9aed0ec6587c8f4a4b3bdc5847355307600d47e4f0cb56c39fed00a",
      "JPKREF":       {
         "SHA256":
"9a0af52ebbb1f1c60fd55e4a26d9a485343c8697457239cc7256350f5bc55071",
         "JPKID": "001000000000000022"
      },
      "verification": "OK"
   },
   "001000000000000022":    {
      "SHA256": "9a0af52ebbb1f1c60fd55e4a26d9a485343c8697457239cc7256350f5bc55071",
      "JPKREF":       {
         "SHA256":
"813a8e56e792155c5f900945c42cffd51a07786babefa90e6ffde3554c686bd8",
         "JPKID": "001000000000000021"
      },
      "verification": "OK"
   },
   "001000000000000021":    {
      "SHA256": "813a8e56e792155c5f900945c42cffd51a07786babefa90e6ffde3554c686bd8",
      "JPKREF":       {
         "SHA256":
"cfb41f58a3998c2c5998609a27ffb84675b5d2b90c947182a6a110cc89848f1e",
         "JPKID": "001000000000000018"
      },
      "verification": "OK"
   },
   "001000000000000018":    {
      "SHA256": "cfb41f58a3998c2c5998609a27ffb84675b5d2b90c947182a6a110cc89848f1e",
      "JPKREF":       {
         "SHA256":
"666fc7bab6a08ee2027ca66de4462ed737adef7a53e87a67427a604eff2426f5",
         "JPKID": "001000000000000012"
      },
      "verification": "OK"
   },
   "001000000000000012":    {
      "SHA256": "666fc7bab6a08ee2027ca66de4462ed737adef7a53e87a67427a604eff2426f5",
      "JPKREF":       {
         "SHA256":
"d91c16165588f5662f58b0980d713843cc8386797451348b69563e628020d76d",
         "JPKID": "001000000000000011"
```

```
      },
      "verification": "OK"
   },
   "00100000000000011":    {
      "SHA256": "d91c16165588f5662f58b0980d713843cc8386797451348b69563e628020d76d",
      "JPKREF":       {
         "SHA256":
"128fefee0a3735c941caa08a1825f85e333f2dfa77cecb63ceaa39715fd464b0",
         "JPKID": "00100000000000010"
      },
      "verification": "OK"
   },
   "00100000000000010":    {
      "SHA256": "128fefee0a3735c941caa08a1825f85e333f2dfa77cecb63ceaa39715fd464b0",
      "JPKREF":       {
         "SHA256":
"20bd55fcf040901b8823099d86874d31094d65a7e3eebfbcbd08965ced225f01",
         "JPKID": "00100000000000009"
      },
      "verification": "OK"
   },
   "00100000000000009":    {
      "SHA256": "20bd55fcf040901b8823099d86874d31094d65a7e3eebfbcbd08965ced225f01",
      "JPKREF":       {
         "SHA256":
"96a249bc10ee9a11e4597069b896c1733ef03db77741373711feea4a9d4c83c2",
         "JPKID": "00100000000000008"
      },
      "verification": "OK"
   },
   "00100000000000008":    {
      "SHA256": "96a249bc10ee9a11e4597069b896c1733ef03db77741373711feea4a9d4c83c2",
      "JPKREF":       {
         "SHA256":
"2ee3ef88b58cc448b4a07c028065dc5e158b4c8d5b4317a3e5407cde2cafe3bd",
         "JPKID": "00100000000000007"
      },
      "verification": "OK"
   },
   "00100000000000007":    {
      "SHA256": "2ee3ef88b58cc448b4a07c028065dc5e158b4c8d5b4317a3e5407cde2cafe3bd",
      "JPKREF":       {
         "SHA256":
"2605674e9de11029d98e41efdf33488a40ad2c875c58cb730e7a903f7f14985f",
         "JPKID": "00100000000000006"
      },
      "verification": "OK"
   },
   "00100000000000006":    {
      "SHA256": "2605674e9de11029d98e41efdf33488a40ad2c875c58cb730e7a903f7f14985f",
      "JPKREF":       {
         "SHA256":
"0000000000000000000000000000000000000000000000000000000000000000",
         "JPKID": "00000000000000000"
      },
      "verification": "OK"
   }
}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON typu:


```
{
```

```
   "code": "B13",
   "message": opis błędu
}
```

W przypadku braku dokumentu o podanym JPKID , zwracany jest JSON:

```
   {
      "code": "B13",
      "message": "Blad: 404"
   }
```

W przypadku podania JPKID niebędącego paragonem, zwracany jest JSON:

```
   {
      "001000000000000030": {"verification": " Not paragon (paragAnul)"}
   }
```

gdzie w nawiasie okrągłym podano typ dokumentu (inny niż spodziewany paragon).

13.3 Operacja weryfikacji ciągłości łańcucha paragonów anulowanych

Wywołanie:

Wywołanie	**GET /api/SerwerCPD/DocumentChain**
Nazwa parametru	**Opis**
- idKasy	Numer unikatowy urządzenia fiskalnego
- startJPKID	Numer początkowy JPKID, od którego będą pobierane kolejne dokumenty w porządku malejącym.
- docType	Typ dokumentu, jednocześnie określenie rodzaju operacji. Dla operacji weryfikacji ciągłości łańcucha paragonów anulowanych `docType=paragAnul`

Odpowiedź:

Poprawna odpowiedź (HTTP/1.1 200 OK Content-Type: application/json) zawiera listę w porządku malejącym (wg JPKID), ograniczoną do maksymalnie 20 pozycji w układzie:

```
"JPKID":{
   "podpis":{
         "RSA": wartość,
         "SHA": wartość,
          "JPK": wartość
    },
    "verification": wynik weryfikacji
}
```

gdzie:

`JPKID` – identyfikator dokumentu, element zawierający:

`podpis` – element zawierający:

`RSA` – wartość podpisu dokumentu,

`SHA` - wartość funkcji skrótu dokumentu,

`JPK` - identyfikator poprzedniego dokumentu,

`verification` – wynik weryfikacji dokumentu, jeśli poprawny, to „OK” w przeciwnym wypadku „Error” i dalsze sprawdzanie jest przerywane.

Dokładny opis tworzenia poszczególnych pól dostępny jest w dokumencie „Opis techniczny protokołu komunikacyjnego kasa – Centralne Repozytorium Kas – Standardy kryptograficzne” w rozdziale „A.10.1d Podpisywanie paragonu anulowanego”.

Przykład wywołania:

https://esb-te.mf.gov.pl:5067/api/SerwerCPD/DocumentChain?idKasy=WTE2001000008&startJPKID=001000000000000030&docType=paragAnul

Przykład odpowiedzi:


```
{
   "001000000000000030":    {
      "podpis":       {
         "RSA":
"30122ba33410bdd399343f9192bfe66bf23c105403e558ff0ae145b72dd83175a15d9aa9b40c2091ba9
282119f908daf0774aaa68219a4930fed5fd4c1bb343afc4fc44a36720c45cb6fe558774eaf178245fc8
5bec702289810e3b82eb6b38e866dc4f47128496fedc9f94930dcc50ad9d519a9dbb8b61c4debb84bef0
b5886c43c3a36e8eb18a75a854e888713f4ebed5f3caf5ca3ff22a8d9f64bf72143bd0748601a6e65496
00f90bf0ec82f6e39b37d275332fb71c66632b7a780c8362516edf5828440274051a4e5c70a7b25c40f6
ed90e12377025fd70c763cd1e91a60d848178643aa2c781b6f5c74f6897eae1ad0c922b56d0dbb7b61ad
8773b7cd2",
         "SHA": "b6af63a56817b77a1dc2739ad2a828d077b76f65ede0ffd94790e6bec4106c52",
         "JPK": "001000000000000024"
      },
      "verification": "OK"
   },
   "001000000000000024":    {
      "podpis":       {
         "RSA":
"214ea96917606b6b1e55e929619e58dee0c2ace64a3292da590e1b8f1614170a6733572ca9d8e0a0c3c
655d4bdfb707f7392d6b51b98327a85de3bfb0bbf8d8331a0890f3df99c57a2534b80f26bb39bc687b01
1c1eb95d47eab2012a5507d0a6bc940760453da9f0fbe5e7ee20669b21f0cad512f4748fba5bcb435247
bbf7724c27dc47958c0fcae964ae1b2990b1c76dfb36ade151bd211a3b9e8d5b29f8c102a5d3fa6a5c11
e523e3110c02a69dc3e060d7628b475696b6005ae53efa4fab1e7e2d9b15672457b112d6b943a5dd64bf
823a9c962d68477876bbb064d7245337b43c4227ad666c1820a7e20dfd24c8938bf417b4c2da9212c9c2
38307addb",
         "SHA": "30fe57f05dd4b6b3744c8cadfb37cd75a7dbb7c3a52eaba4f17aab9fbd1ed4e7",
         "JPK": "001000000000000020"
      },
      "verification": "OK"
   },
   "001000000000000020":    {
      "podpis":       {
         "RSA":
"1113a835625f5ace5ff62caafc423555a91087c580bc903b2d9ac1074184228852c526dcf898a2c9713
7d84f21ebd95109ed5fead30a1de6005b255ca6ef64d8de8d99336eb0208e4e315ccc8806e56d590faf2
3db6a9e28efd42e0d9ebd509ba8517daa6448f83a0b65a41fcc238052e2a99684dacaa431e74912e27fe
142386b4d3633223f7c1333e5fb52ed48a94b9e0e14322f83148b97685e54cf9b280230df3dce3b98313
4accec79ea90c06f4d09f41eb9c3d62c00d18ea6b227ded9d67c1d6c215a0b9c52e7d256a0e8bd573f77
5cd76baba709c5260f00acf9008bbdaae291f17d13c5d5e81ee23e18b9534e78ff34bed4e38bcbeb422d
83a62cf7c",
         "SHA": "e1e1110170c4bf576779c57d810a5c8c8cfc4f76aed6eef604130b37ece3bdc5",
         "JPK": "001000000000000019"
      },
      "verification": "OK"
   },
   "001000000000000019":    {
      "podpis":       {
         "RSA":
"410a5e00a247581e17861aa42e81404cbfcc048611e68549e7433824df13d13bd639900240437fa58ca
0e80c6531806f007edfc86272a04d74e52bcf8d729c23c7a8122ba81a85b85a5b42fe65301b47434aca9
8b4de380886013e1ea1ddb3d7b49dee3f1e7b04e81dbc322a17343e74b6d33bad7767dcb58c5b50b0dd2
58f50c79f1ec73e718e0753ae382c0ed65058641d39bcd39208332e0beb0ad7c7f4db1bfd5551f1874e5
affd705d5457cceab9644c193eabb95414b1d498b11c18a307e6ddb65bbf6fb3a7352b7cca789953ce3a
```

```
de07c3cdc1dad8cb44967d03c38714e7f5bf2c68c901577632de08b6785670cfeb38244c7d0cee19a426
288de5030",
         "SHA": "eaa222bba4cd7bdb2655847b85ad502f116f7031226d2a1535feb9d974d26ab9",
         "JPK": "001000000000000015"
      },
      "verification": "OK"
   },
   "001000000000000015":    {
      "podpis":       {
         "RSA":
"2fcd1742b01d1a2f17b71a098026b48cf50716d0a778e281ed56f9980ea7bcba4cd5fcefed3fc2c63b6
bea160344f53803190fe1d805a94aaa13d9a9d539312f2a50eb079ca59becff97a87711fedc5173769fb
41ca72eb99703aab63516db9a60bff298d37673eaf3214475fa27f44722bd218f058c52e452cb1e286e5
3a7d1d5df76495513a361e0e82f807d24b030fa48ed7623788670f28b47886ac7fccb7ff786dc454d471
42e0ad813b8b83f27aa5e072c9fc6639d33478a26f73fd8a3ad1a33880ae9ebeac8efd06e72a35f79627
e0f0e3b5c0670b8104db47caa63eac4f25823eaa835c807f5c73e3539e9bda775e6d1b6431a8048bd5a0
6d6595027",
         "SHA": "089f3a946b96c2fbb051ed5d0d70d97c0fdaae592f37f61b086c17eb0f84e3a9",
         "JPK": "001000000000000013"
      },
      "verification": "OK"
   },
   "001000000000000013":    {
      "podpis":       {
         "RSA":
"54631e9611bc2de38de5108cdf085942c2fc448c12b066dd1a717002d0629c69fb2415fb28a2eaa2e6a
c6a977610859b88c68bdd8457369d45fece190668fab487f6456ce1c52a61f8efcb432a5b8f8b2fcedd2
d73f5f27526062f531dea42538303b47756428b55056ea383e192ff59b3730835ef1799de27bfdfb37f0
fd24a144b59b92e2f235b50f4c2037d975ae8fe5f7e9df6381470fe9e369b2b100bd4b340f1cd9f2f9ba
df2c9378eea8bad3dbc13997775a6e98cf5435cc2e05c57c85dc750c75e543dd1da7da787b7f82b7c834
a8ac30090d82aa700da1fe3f18805476807284a1202bcce58749a382218920239bcd62edd871c03883a6
b7c58ec07",
         "SHA": "09d3becac04391854f3d5329999c18cf8e425ea1156bffb88ed539e9ff4524fd",
         "JPK": "001000000000000005"
      },
      "verification": "OK"
   },
   "001000000000000005":    {
      "podpis":       {
         "RSA":
"64703a7aaf872b073037397c7c9982a22416f6311f7766a81067d13703ad894595696bd85b469c7c6bf
064c3e09d1832ed81b9c8e762f2919e821c444446ea3dbef8d3c118cbbb3611b3128cfd9bc9c33377b9e
118170c75c6b21c9cd853a9cc3e67194052fb4b6e294717359f8f3a2a7224f1a67bce1de39eec420e482
4688ffed4fe71b2efad0c53594e5d78e8906faedda423501ac21fb0004ec75d1448db5cddc566d32fa15
501f793ec285b42f631a9b2496b2301ce42e9c58ab1004a79d7d90c4d7fbadde8d210259c33854b7d75b
13c03f896181e7861582d1f897a8b1b6ade2f9bc20419e371eb260dbbb60f1d65eb6b178d8cef126a8ca
6f72505b2",
         "SHA": "c054507245b073be930aaadf337a3922a72c6ac09cd2ecf0fee4c690949e8172",
         "JPK": "001000000000000004"
      },
      "verification": "OK"
   },
   "001000000000000004":    {
      "podpis":       {
         "RSA":
"86c4eadeb5010f9d38d846464875ee8570ba6c11c579e54c765617d6ead9bc0804bc0bf4bfabe1e0721
d31ec169b67f00dc07d489e79695ad5c2f3f984c2cd2be4a568bf982a0b2d169fc14960a89c3484853a4
e924c5326dc50e6c0411991fe5c373561760f3eac5768c0e7abfb4a94682f194b3464abd8fddbc473394
badb9c562d3df25107c79f03ea55e23a48f5b5835f2f2fcdcd4de0ab0b2a1fd48b81e63593a8a81c28a6
26876fc630b711a212176716a464bd87fad6e1ca30c2db0e0443b0fa4adb525bd5f0c3b498005992c552
```

```
bcdd8845439b1b2ce05ce0ad7bebf1bd1be9c0aee7c679e5025d818ba86f5d13f1882fb9027051a403f9
cf5e03732",
         "SHA": "dbc80c2751df6c3491e873c2d9ddae82c1a930ec6e13592f7f7c8409e761ac75",
         "JPK": "001000000000000003"
      },
      "verification": "OK"
   },
   "001000000000000003":    {
      "podpis":       {
         "RSA":
"39d8ae371092541f7af5f7b66e0b5588fec3d0947592e39b83d1578378055a3afef9216ae22ea687277
c07fd0be6e85170f4740ed499a4c827e76866c0eca38671688429280a68fac10730c711d202bb1e78ac5
e271a30d73ff1439ae64aef0a59b3bd73a36cbbeec0ec8a66762e603fb679cee081866d43a54d8c480ba
ba859d09db9acbccfb82cba2c558e2b21afa6535b04218f06da2ca3c1f6da56381068d701d5f59c1c14c
a81e334120d8373df68128667ade85d0d588f1faeb32b625f11e2d115073792c22905f1d680bafc36f0c
1d5c744ea7e45c9b62ff4ff1cfecbd44f851ab6e44550f7a8f01aa8d040d95f7832e4307e6dd5317e38a
bdbe44358",
         "SHA": "8214e92f4fcac131ea16fe3fb54bd4ee8c1898ec3ba8c9302a93e5f0ebac6a21",
         "JPK": "001000000000000002"
      },
      "verification": "OK"
   },
   "001000000000000002":    {
      "podpis":       {
         "RSA":
"0b8165743a8265eaf29c7ca56df70b9a00709121dd8b60d9ea43ab0a27e3251b5d3a0c1400a3f81a286
17d08da384c03e984e42a951ab5b26cb2d6e2ad20ee5aa998cc35053fcd209fc0fdee05299021e1c9d9e
4921fa39bd67e97dc1ebaf105baad1055880325d33852dff6620930a44a2dedbc4f9a760b0165ab26aea
103b872a38aa7be67d65d08d76a4a57f9fca1073097d33e6338bab3cf76687afbd72e607c1e74c36429e
b46cd907f893a507dda0ee86b22761c5c24af7836d4b75c6c7f6110cce526457c872e3685520b7cf62ff
89fad236322811f1829c517880cb3d0d5a1d97897bc449c37f88e84ee50b991ccd4e1e4c8716eca03497
15893219d",
         "SHA": "00c1641f9f18a3c3a1f7d5e2611136a847b763858af7d7f2fc110b537c2b34cc",
         "JPK": "000000000000000000"
      },
      "verification": "OK"
   }
}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON typu:

```
   {
      "code": "B13",
      "message": opis błędu
   }
```

W przypadku braku dokumentu o podanym JPKID , zwracany jest JSON:

```
   {
      "code": "B13",
      "message": "Blad: 404"
   }
```

W przypadku podania JPKID niebędącego paragonem, zwracany jest JSON:

```
   {
      "001000000000000031": {"verification": " Not paragAnul (paragon)"}
   }
```

gdzie w nawiasie okrągłym podano typ dokumentu (inny niż spodziewany paragAnul).

13.1 Operacja weryfikacji ciągłości łańcucha raportów dobowych

Wywołanie:

Wywołanie	**GET /api/SerwerCPD/DocumentChain**
Nazwa parametru	**Opis**
- idKasy	Numer unikatowy urządzenia fiskalnego
- startJPKID	Numer początkowy JPKID, od którego będą pobierane kolejne dokumenty w porządku malejącym.
- docType	Typ dokumentu, jednocześnie określenie rodzaju operacji. Dla operacji weryfikacji ciągłości łańcucha paragonów anulowanych `docType=rapDob`

Odpowiedź:
Poprawna odpowiedź (HTTP/1.1 200 OK Content-Type: application/json) zawiera listę w porządku malejącym (wg JPKID), ograniczoną do maksymalnie 20 pozycji w układzie:

```
"JPKID":{
   "podpis":{
         "RSA": wartość,
         "SHA": wartość,
          "JPK": wartość
    },
    "verification": wynik weryfikacji
}
```

gdzie:
`JPKID` – identyfikator dokumentu, element zawierający:
`podpis` – element zawierający:
`RSA` – wartość podpisu dokumentu,
`SHA` - wartość funkcji skrótu dokumentu,
`JPK` - identyfikator poprzedniego dokumentu,
`verification` – wynik weryfikacji dokumentu, jeśli poprawny, to „OK” w przeciwnym wypadku „Error” i dalsze sprawdzanie jest przerywane.

Dokładny opis tworzenia poszczególnych pól dostępny jest w dokumencie „Opis techniczny protokołu komunikacyjnego kasa – Centralne Repozytorium Kas – Standardy kryptograficzne” w rozdziale „A.10.1b Podpisywanie raportu fiskalnego dobowego”.

Przykład wywołania:
https://esb-te.mf.gov.pl:5067/api/SerwerCPD/DocumentChain?idKasy=WTE2001000008&startJPKID=0010000000000000029&docType=rapDob

Przykład odpowiedzi:

```
{
   "00100000000000029":    {
      "podpis":       {
         "RSA":
"5295572bc4dd551a118c56369a5da5af3d6ec74c4474a46eab637805647f7ac01dae4ba2f155438cfce
4b4e7049841b74586079dca71a8a37c01f35c9185c45f7df80ff4edd380410b5cfc0974c9a3807449a92
09295f9c7728f68d3e04a0e0b54cc8724ef9541ef58eb371d66bd4af3612261501544750495af0de38ae
37254cc191f9cb5d80ef9b9b37bd4dc5000ad424a6f8f84e95c1c676374ce3fcd9b14741821936577276
0ebcb07f0bc22cbbe276a21c982d7353923b1065df5f1835b3450d4a55ac14db67ae176b1479b6485d6a
0ef9fa943c72e94f0a22bef1abb4fc3804e750bc63b8eeaab75d6535ae1175b6c3988fb8eeb08f626558
9517e39f8",
         "SHA": "8cb340afea6b4f2995e350a6257228ad93a0ed7334fa56e1ade93c7df6e63272",
         "JPK": "00100000000000017"
      },
      "verification": "OK"
   },
   "00100000000000017":    {
```

```
      "podpis":       {
         "RSA":
"1c5a35dfd7082919addadd1b392a8134a0eac9c79ad247280d2991e2521cfee9d4ef1bcf03ce54fc071
ba914b8f1dc95660b7f594591a2ca2349d8267f30af520a43fba8e72cfd330a2c47745ea6b3db0a2c5b2
cfd67f7955dc8e1cb199db879e7121602558c36e3b966a9335085bfa10568c18605fe8f1b2089a3e3175
835482c71e35c564805a7c02574a046a99efeb4d10b7e6d4e3f3db724022e148a6cbb06dfbab2e002169
01a8d79a80b4b86d496fcd415b27721e2a8afb06d40e24f06b976fe6bb628220d09d48ca971b499949fe
4ef58409534d1770800e483afa4d1932bff1d2ec17485d209d6dd1ac955bee3baf15a9c4d8450a2252d9
beda460db",
         "SHA": "e497345505d8de57fa5196ca815e237a8c70b3107d57430fb4604aa9c605764e",
         "JPK": "000000000000000000"
      },
      "verification": "OK"
   }
}
```


W przypadku błędu w parametrze lub błędu podczas wykonywania usługi, zwracany jest JSON typu:

```
   {
      "code": "B13",
      "message": opis błędu
   }
```

W przypadku braku dokumentu o podanym JPKID , zwracany jest JSON:

```
   {
      "code": "B13",
      "message": "Blad: 404"
   }
```

W przypadku podania JPKID niebędącego paragonem, zwracany jest JSON:

```
   {
      "00100000000000031": {"verification": " Not rapDob (paragon)"}
   }
```

gdzie w nawiasie okrągłym podano typ dokumentu (inny niż spodziewany rapDob).