Специфікація
WEB API ПРРО Cashälot



версія 2.1.0.8034 від 29.09.2026



Зміст

Загальні положення 4
Матеріали для ознайомлення 4
Необхідні доступи 4
Перед початком роботи 4
Обмін між обліковою системою і WEB API 5
Автоматичне завершення режиму офлайн 5
Використання WEB API 6
Команди 7
Загальні параметри запиту 7
Використання PIN та ідентифікатора ключа (KeyGuid) 7
Загальні параметри відповіді 7
Запит стану сервера: ServerState 7
Запит доступних об'єктів: Objects 8
Запит переліку операторів (касирів) для суб’єкта господарювання: Operators 9
Запит стану ПРРО: TransactionsRegistrarState 9
Відкриття зміни: OpenShift 10
Закриття зміни: CloseShift 11
Службове внесення: RegisterCheck 12
Службова видача: RegisterCheck 13
Реєстрація фіскального чека: RegisterCheck 14
Реєстрація Z-звіту: RegisterZRep 16
Запит підсумків останньої зміни: LastShiftTotals 17
Встановлення залишку готівки на кінець попередньої зміни: SetCashInHandPrevShift 18
Приведення локального стану ПРРО у відповідність до стану ПРРО на Фіскальному Сервері: Cleanup 18
Запит чека з сервера: GetCheck 19
Запит Z-звіту з сервера: GetZRep 20
Запит переліку змін за період: Shifts 21
Запит переліку документів зміни: Documents 22
Запит відомостей про документ за локальним номером: DocumentInfoByLocalNum 23
Налаштування ПРРО: SetupRegistrar 23
Запит налаштувань ПРРО: RegistrarSettings 25
Запит інформації сертифіката: CertificateInfo 26
Отримання переліку сертифікатів з контейнера у форматі PFX/JKS: ContainerCertificates 26
Конвертування XML документа у форматі НБУ: ConvertDocument 27
Надсилання чека споживачу: SendCheckToConsumer 27
Структури даних 28
Фіскальний чек 28
Z-звіт 32
Константи 35

Клас документа 35
Чек. Тип документа 35
Чек. Розширений тип документа 36
Чек. Тип часткової сплати 37
Тип даних запита документа 37
Режим роботи ПРРО 38
Код результату обробки запиту документа 38
Сервіси надсилання повідомлень споживачу 39
Підтримка документів у форматі НБУ 40
Приклади документів у форматі НБУ 40
Фіскальний чек 40
Z-звіт 41

Загальні положення

Модуль API взаємодії з фіскальним сервером програмних реєстраторів розрахункових операцій (далі Модуль API) призначений для спрощення взаємодії програмного забезпечення Облікової Системи Користувача (ОСК) і Фіскального Сервера контролюючого органу (ФСКО).

API забезпечує:

- обмін з обліковою системою за протоколом HTTPS в форматі JSON;
- створення документів (чеків, Z-звітів) в форматі XML для надсилання на ФСКО;
- засвідчення документів кваліфікованим електронним підписом (КЕП);
- розрахунок підсумків по зареєстрованим чекам;
- автоматичне створення Z-звітів на підставі розрахованих підсумків;
- автоматичний перехід в режим офлайн;
- автоматичне повернення в режим онлайн з реєстрацією на ФСКО документів, створених в режимі офлайн;
- використання програмних (файлових) ключів КЕП, а також хмарного підпису (Smart ID).

Матеріали для ознайомлення

Перед початком роботи з WEB API Cashalot, необхідно ознайомитись з матеріалами:

1. Опис API фіскального сервера (Єдине вікно подання електронної звітності)
2. Опис структур документів.
3. Приклади документів.

Посилання: https://tax.gov.ua/baneryi/programni-rro/opis-ari-fiskalnogo-servera/opis-ari--kontrolyuyuchogo-organu/

Необхідні доступи

1. Доступ до кабінету користувача – https://my.cashalot.org.ua
2. Протоколи TLS 1.2 і TLS 1.3.
3. Адреса відправки запитів – https://fsapi.cashalot.org.ua/

Перед початком роботи

1. Потрібно зареєструватись в кабінеті користувача - https://my.cashalot.org.ua/
2. В кабінеті користувача потрібно зареєструвати ГО, каси та касирів.
3. В списку кас необхідно завантажити та зберегти ліцензії для ПРРО.

Обмін між обліковою системою і WEB API

1. Обмін між обліковою системою і WEB API здійснюється за протоколом HTTPS.
2. Тип HTTPS-запиту – POST.
3. Заголовок "Content-Type": "application/json" чи "Content-Type": "application/json; charset=UTF-8".
4. Використання компресії Gnu Zip (GZip) або Zip (deflate). Вхідні та вихідні документи обробляються алгоритмом GZip або Zip для зменшення розміру повідомлення.
5. У разі використання компресії, заголовок "Content-Encoding" вхідного повідомлення повинен мати значення "gzip" або "deflate".
6. Формат повідомлень ОСК і відповідей Модуля API – JSON (https://www.json.org/).

Автоматичне завершення режиму офлайн

Якщо фіскальний сервер став доступним в режимі онлайн завершення режиму роботи офлайн і повернення до режиму роботи онлайн здійснюється для кожного ПРРО окремо, під час обробки команд "Запит стану ПРРО", "Реєстрація фіскального чека" тощо.

Використання WEB API

1. Початок роботи ОСК з ПРРО.
   У разі необхідності ОСК надсилає запит доступних об'єктів "Objects".
   На підставі відповіді визначається доступність ПРРО для роботи.
   Примітка. Якщо для ПРРО параметр "Closed" = "true", дозволяється тільки реєстрація Z-звіту і закриття зміни.
2. У разі необхідності ОСК надсилає запит стану ПРРО "TransactionsRegistrarState".
   Якщо зазначений ПРРО раніше не використовувався із Модулем API, тобто команда "TransactionsRegistrarState" надсилається для ПРРО вперше, необхідно, щоб ФСКО знаходився в режимі онлайн. У протилежному випадку Модуль API не може отримати необхідну інформацію щодо поточного стану ПРРО.
3. Для початку зміни ОСК надсилає запит відкриття зміни "OpenShift".
4. Для завершення зміни ОСК надсилає запит закриття зміни "CloseShift". Якщо Z-звіт не було зареєстровано, параметр "ZRepAuto" дозволяє автоматичне створення і реєстрацію Z-звіту.
5. Для реєстрації фіскального чека ОСК надсилає запит "RegisterCheck". Якщо реєструється сторнуючий чек ("CHECKHEAD.DOCSUBTYPE" = "CheckStorno"), для коректного розрахунку підсумків необхідно зазначити дані фіскального чека, що сторнується, в параметрі "StornedCheck".
6. У разі необхідності реєстрації Z-звіту окремо від завершення зміни ОСК надсилає запит "RegisterZRep". Якщо параметр "ZRep" не зазначений, заповнення Z-звіту виконується автоматично.
7. Для одержання підсумків поточної зміни ОСК надсилає запит "LastShiftTotals".
   Примітка. Для автоматичного створення Z-звіту будуть використані саме ці підсумки.
8. Команду "Cleanup" необхідно використовувати у випадках, коли стан ПРРО в Модулі API відрізняється від фактичного стану ПРРО на ФСКО.
   Про таку ситуацію говорить одержання коду помилки "InconsistentRegistrarState".
   Причиною розбіжностей стану може стати некоректне використання ПРРО, наприклад одночасна реєстрація документів ПРРО на декількох пристроях.
   Видаляються всі збережені локально дані ПРРО, зокрема документи, створені в режимі офлайн і не надіслані на ФСКО.
   Якщо зміну ПРРО на ФСКО відкрито, зміна закривається з автоматичним створенням Z-звіту у разі необхідності.

Команди

Загальні параметри запиту

Всі команди, за виключенням "Запит стану сервера" (ServerState), "Запит чека з сервера" (GetCheck) і "Налаштування ПРРО" (SetupRegistrar) і “Конвертування XML документа у форматі НБУ” (ConvertDocument), повинні містити параметри:

- "UID": "Не обов’язковий унікальний ідентифікатор запиту (null – не використовується)"
- "Certificate": "Сертифікат підписувача в кодуванні Base64"
- "PrivateKey": "Секретний ключ підписувача в кодуванні Base64 (null-використовується хмарний підпис (Smart ID)) "
- "Password": "Пароль секретного ключа підписувача"
- "UseSmartId": "Використовувати хмарний підпис (Smart ID)" (false/true)

Використання PIN та ідентифікатора ключа (KeyGuid)

У випадках, коли в кабінеті користувача Cashalot активовано додаткові налаштування Інтеграції з WEB API (розділ Налаштування → Інтеграція з WEB API), система може використовувати PIN та ідентифікатор ключа (KeyGuid) для оптимізації процесів реєстрації розрахункових операцій.

Використання

1. У відповідних командах необхідно:

- не заповнювати поля: “Certificate”, “PrivateKey”, “Password”, “UseSmartId”;
- заповнити поле “KeyGuid” — ідентифікатор ключа (KeyGuid) (використовується, якщо у налаштуваннях кабінету користувача для касира відсутні додані ПРРО);
- заповнити поле “KeyPin” — PIN.

Примітка: ідентифікатор ключа (KeyGuid) та PIN відображаються у кабінеті користувача Cashalot у розділі Налаштування → Інтеграція з WEB API → Управління касирами.

2. У разі використання PIN-коду (через параметр “KeyPin”) для команд:

- “Objects” — Запит доступних об'єктів;
- “Operators” — Запит переліку операторів (касирів) для суб’єкта господарювання

слід передавати параметр:

“NumFiscal” — Фіскальний номер ПРРО (якщо у налаштуваннях кабінету користувача для касира додані ПРРО)

або

“KeyGuid” — Ідентифікатор ключа (KeyGuid) (якщо у налаштуваннях кабінету користувача для касира відсутні додані ПРРО)

Загальні параметри відповіді

- "ErrorCode": "Код помилки"
- "ErrorMessage": "Опис помилки"

Запит стану сервера: *ServerState*

Запит призначений для перевірки доступності ФСКО в режимі онлайн.

Параметри та приклад запиту:

```
{
  "Command": "ServerState"
}
```

Параметри відповіді:

```
{
   "Available":    , // Ознака доступності в режимі онлайн (false/true)
   "Version":      // Номер версії системи
}
```

Запит доступних об'єктів: Objects

Запит повертає перелік господарських одиниць і ПРРО, що зареєстровані на ФСКО і доступні підписувачу.

Параметри та приклад запиту:

```
{
   "Command": "Objects",
   "UID": "2BC34BCA-4664-4488-B1A6-14EBFCF41382", // Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується),
   "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64,
   "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64,
   "Password": "password" // Пароль секретного ключа підписувача
}
```

Параметри відповіді:

```
{
   "TaxObjects": [ // Масив Господарських одиниць
      {
         "Entity": // Ідентифікатор ГО,
         "TaxObjGuid": // Ідентифікатор запису ОО,
         "TaxObjId": // Код запису ОО,
         "SingleTax": // Ознака ФОП - платника єдиного податку,
         "Name": // Найменування ГО,
         "Address": // Адреса ГО,
         "Tin": // Код ЄДРПОУ/ДРФО платника податків,
         "Ipn": // Індивідуальний податковий номер платника ПДВ продавця,
         "OrgName": // Найменування суб’єкта господарювання
         "ChiefCashier": // Старший касир
         "TransactionsRegistrars":  [ // Масив ПРРО, зареєстрованих для ГО
            {
               "NumFiscal": // Фіскальний номер ПРРО
               "NumLocal": // Локальний номер ПРРО
               "Name": // Найменування ПРРО
               "Closed": // Ознака скасованої реєстрації ПРРО, на якому наразі не закрито зміну
            }
         ]
      }
   ]
}
```

Запит переліку операторів (касирів) для суб’єкта господарювання: *Operators*

Запит повертає перелік операторів, що зареєстровані для суб'єкта господарювання, реєстраційний номер якого (ЄДРПОУ, ДРФО, Картка платника податків) міститься у сертифікаті КЕП, яким засвідчений запит.

Параметри та приклад запиту:

```
{
  "Command": "Operators",
  "UID": "6DE503EC-A7A6-4E9B-88FF-B1CE77E2D4D5", // Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password" // Пароль секретного ключа підписувача
}
```

Параметри відповіді:

```
{
  "Tin": // Реєстраційний номер суб'єкта господарювання (ЄДРПОУ, ДРФО, Картка платника податків),
  "Operators": [ // Масив переліку операторів
    {
      "SubjectKeyId": // Ідентифікатор ключа суб’єкта,
      "RegNum": // Реєстраційний номер особи оператора (ЄДРПОУ, ДРФО, Картка платника податків),
      "ChiefCashier": // Старший касир (true/false)
    }
  ]
}
```

Запит стану ПРРО: *TransactionsRegistrarState*

Запит повертає параметри поточного стану ПРРО.

Параметри та приклад запиту:

```
{
  "Command": "TransactionsRegistrarState",
  "UID": "758D7562-08C3-49C9-A023-634DB599B126", // Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "IncludeTaxObject": true // Ознака запиту відомостей об’єкту оподаткування (не обов’язковий) (false/true)
}
```

Параметри відповіді:

```
{
  "ShiftState": // 0-зміну не відкрито, 1-зміну відкрито
  "ShiftId": // Ідентифікатор зміни
  "OpenShiftFiscalNum": // Фіскальний номер документа "Відкриття зміни"
  "ZRepPresent": // Ознака присутності Z-звіту (false/true)
  "Testing": // Ознака зміни, що містить тестові документи (false/true)
  "Name": // П.І.Б. оператора, що відкрив зміну
  "SubjectKeyId": // Ідентифікатор ключа суб'єкта сертифіката оператора, що відкрив зміну
  "FirstLocalNum": // Перший внутрішній номер документа у зміні
  "NextLocalNum": // Наступний внутрішній номер документа
  "LastFiscalNum": // Останній фіскальний номер документа
  "OfflineSupported": // Для ПРРО може використовуватись режим офлайн
  "ChiefCashier": // Користувач є старшим касиром
  "OfflineSessionId": // Ідентифікатор офлайн сесії (null – режим офлайн заборонений для ПРРО)
  "OfflineSeed": // "Секретне число" для обчислення фіскального номера офлайн документа (null – режим офлайн заборонений для ПРРО)
  "OfflineNextLocalNum": // Наступний локальний номер документа в офлайн сесії (null – режим офлайн заборонений для ПРРО)
  "OfflineSessionDuration": // Тривалість офлайн сесії (хвилин) (null – режим офлайн заборонений для ПРРО)
  "OfflineSessionsMonthlyDuration": // Сумарна тривалість офлайн сесій за поточний місяць (хвилин) (null – режим офлайн заборонений для ПРРО)
  "Closed": // Ознака скасованої реєстрації ПРРО, на якому наразі не закрито зміну
  "OfflineDocumentsPresent": // Ознака наявності офлайн документів, що чекають надсилання на фіскальний сервер (false/true)
  "OfflineDocumentsAutoRegState": // Статус автоматичної реєстрації офлайн документів" (null – помилки відсутні)
  "AutoCloseShiftState": // Статус автоматичного закриття зміни (null – помилки відсутні)
  "LicenseExpiration": // Дата завершення ліцензії
  "TaxObject": // Відомості об'єкту оподаткування (TaxObjectItem)
}
```

Відкриття зміни: *OpenShift*

Запит відкриває зміну на ПРРО.

Параметри та приклад запиту:

```
{
  "Command": "OpenShift",
  "UID": "44A5A6B2-4770-417B-8E7B-750C6CDB7627", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890 // Фіскальний номер ПРРО
}
```

Параметри відповіді:

```
{
   "NumFiscal": // Фіскальний номер документа
   "NumLocal": // Локальний номер документа
   "OrderDateTime": // Дата і час документа
   "Offline": // Документ зареєстровано в режимі офлайн (false/true)
}
```

Закриття зміни: *CloseShift*

Запит закриває зміну на ПРРО.

Параметри та приклад запиту:

```
{
   "Command": "CloseShift",
   "UID": "35852E40-D3DB-4337-9326-85177C7ED930", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
   "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
   "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
   "Password": "password", // Пароль секретного ключа підписувача
   "NumFiscal": 1234567890, // Фіскальний номер ПРРО
   "ZRepAuto": true, // Автоматично створювати Z-звіт, якщо відсутній (false/true)
   "Visualization": true, // Надати візуалізацію документа (автоматично створеного Z-звіту) (false/true)
   "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
   "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
   "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
   "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
   "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Примітка. Якщо жоден параметр "VisAs…" не має значення "true", документ виводиться у текстовому UTF-8 форматі.

Параметри відповіді:

```
{
   "ZRepAutoInfo": { // Відомості автоматично створеного Z-звіту
      …
   },
   "NumFiscal": // Фіскальний номер документа
   "NumLocal": // Локальний номер документа
   "OrderDateTime": // Дата і час документа
   "Offline": // Документ зареєстровано в режимі офлайн (false/true)
   "Visualization": // Документ (автоматично створений Z-звіт) у вибраному форматі (текст UTF-8,HTML, PDF або PNG) в кодуванні Base64
}
```

Службове внесення: *RegisterCheck*

Параметри та приклад запиту:

```
{
  "Command": "RegisterCheck",
  "UID": "788B0999-59EC-4CC3-89C5-D7F69956D3FB", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "Check": {
    "CHECKHEAD": {
      "DOCTYPE": "SaleGoods",
      "DOCSUBTYPE": "ServiceDeposit"
    },
    "CHECKTOTAL": {
      "SUM": 1000
    }
  },
  "GetQrCode": true, // Одержати QR-код посилання на чек в "Електронному кабінеті платника податків" (false/true)
  "Visualization": true, // Надати візуалізацію документа (false/true)
  "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
  "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
  "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
  "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
  "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Примітка. Якщо жоден параметр "VisAs…" не має значення "true", документ виводиться у текстовому UTF-8 форматі.

Приклад відповіді:

```
{
  "QrCode": // QR-код посилання на чек в "Електронному кабінеті платника податків" в форматі "JPEG" в кодуванні Base64
  "Url": // URL посилання на чек в "Електронному кабінеті платника податків"
  "NumFiscal": // Фіскальний номер документа
  "NumLocal": // Локальний номер документа
  "OrderDateTime": // Дата і час документа
  "Offline": // Документ зареєстровано в режимі офлайн (false/true)
  "Visualization": // Документ (автоматично створений Z-звіт) у вибраному форматі (текст UTF-8,HTML, PDF або PNG) в кодуванні Base64
}
```

Службова видача: *RegisterCheck*

Параметри та приклад запиту:

```
{
   "Command": "RegisterCheck",
    "UID": "F18A16EA-1744-4150-BC31-936A74D94C5F", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
   "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
   "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
   "Password": "password", // Пароль секретного ключа підписувача
   "NumFiscal": 1234567890, // Фіскальний номер ПРРО
   "Check": {
      "CHECKHEAD": {
         "DOCTYPE": "SaleGoods",
         "DOCSUBTYPE": "ServiceIssue"
      },
      "CHECKTOTAL": {
         "SUM": 100
      }
   },
   "GetQrCode": true, // Одержати QR-код посилання на чек в "Електронному кабінеті платника податків" (false/true)
   "Visualization": true, // Надати візуалізацію документа (false/true)
   "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
   "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
   "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
   "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
   "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Примітка. Якщо жоден параметр "VisAs…" не має значення "true", документ виводиться у текстовому UTF-8 форматі.

Приклад відповіді:

```
{
   "QrCode": // QR-код посилання на чек в "Електронному кабінеті платника податків" в форматі "JPEG" в кодуванні Base64
   "Url": // URL посилання на чек в "Електронному кабінеті платника податків"
   "NumFiscal": // Фіскальний номер документа
   "NumLocal": // Локальний номер документа
   "OrderDateTime": // Дата і час документа
   "Offline": // Документ зареєстровано в режимі офлайн (false/true)
   "Visualization": // Документ (автоматично створений Z-звіт) у вибраному форматі (текст UTF-8,HTML, PDF або PNG) в кодуванні Base64
}
```

Реєстрація фіскального чека: *RegisterCheck*

Запит реєструє фіскальний чек на ПРРО.

Параметри запиту:

- "Command": "RegisterCheck"
- "UID": " Унікальний ідентифікатор запиту (null-не використовується)"
- "NumFiscal": "Фіскальний номер ПРРО"
- "Check": "Дані фіскального чека"
- "CheckToConvert": "Дані фіскального чека (XML документ у форматі НБУ в кодуванні Base64)"
- "StornedCheck": "Дані фіскального чека, що сторнується. Використовуються, якщо реєструється чек типу "Чек сторнування попереднього чека" (CheckStorno)"
- "StornedCheckToConvert": "Дані фіскального чека, що сторнується (XML документ у форматі НБУ в кодуванні Base64). Використовуються, якщо реєструється чек типу "Чек сторнування попереднього чека" (CheckStorno)"
- "AutoOpenShift": "Автоматично відкрити зміну, якщо не відкрита" (false/true)
- "GetQrCode": "Одержати QR-код посилання на чек в "Електронному кабінеті платника податків"" (false/true)
- "QrSize": "Бажаний приблизний розмір QR-коду по вертикалі/горизонталі в пікселях (від 256 до 2048, за замовчанням 512)"
- "Visualization": "Надати візуалізацію документа" (false/true)
- "VisAsHtml": "Візуалізація у форматі HTML" (false/true)
- "VisAsPdf": "Візуалізація у форматі PDF" (false/true)
- "VisAsPng": "Візуалізація у форматі PNG" (false/true)
- "VisWidth": "Ширина області візуалізації у пікселях (0 - за замовчанням значення 440, максимальне значення 4096)"
- "VisTransparentBackground": "Прозорий фон візуалізації" (false/true)

Примітка. Якщо зміна відкривається завдяки значенню параметра "AutoOpenShift": true, то ознака відкриття зміни в тестовому режимі визначається параметром "Check.CHECKHEAD.TESTING".

Примітка. Параметр "CheckToConvert" використовується у разі відсутності зазначеного параметра "Check".

Примітка. Параметр "StornedCheckToConvert" використовується у разі відсутності зазначеного параметра "StornedCheck".

Примітка. Якщо жоден параметр "VisAs…" не має значення "true", документ виводиться у текстовому UTF-8 форматі.

Приклад запиту:

```
{
  "Command": "RegisterCheck",
  "UID": "27C04CE9-A8CD-4CD4-A7C2-968DC76D82ED", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
```

```
"NumFiscal": 4000000004, // Фіскальний номер ПРРО
"Check": {
  "CHECKHEAD": {
    "DOCTYPE": "SaleGoods",
    "DOCSUBTYPE": "CheckGoods",
    "COMMENT": "Коментар до чека"
  },
  "CHECKTOTAL": {
    "SUM": 99.99,
    "COMMISSION": 10.00
  },
  "CHECKPAY": [
    {
      "PAYFORMCD": 0,
      "PAYFORMNM": "ГОТІВКА",
      "SUM": 109.99,
      "PROVIDED": 200.00,
      "REMAINS ": 90.01
    }
  ],
  "CHECKTAX": [
    {
      "TYPE": 0,
      "NAME": "ПДВ",
      "LETTER": "А",
      "PRC": 20.00,
      "TURNOVER": 99.99,
      "SOURCESUM": 99.99,
      "SUM": 15.87
    },
    {
      "TYPE": 1,
      "NAME": "Акциз",
      "LETTER": "Г",
      "PRC": 5.00,
      "TURNOVER": 99.99,
      "SOURCESUM": 99.99,
      "SUM": 4.76
    }
  ],
  "CHECKBODY": [
    {
      "CODE": "123456789000",
      "BARCODE": "789005",
      "UKTZED": "2905110000",
      "NAME": "Мет. спирт ЧДА 99.9% USA",
      "UNITCD": 138,
      "UNITNM": "л",
      "AMOUNT": 1.000,
```

```
            "PRICE": 99.99,
            "LETTERS": "АГ",
            "COST": 99.99,
            "COMMENT": "Коментар до товару"
        }
      ]
   },
   "GetQrCode": true,
   "Visualization": true
}
```

Параметри відповіді:

```
{
   "QrCode": // QR-код посилання на чек в "Електронному кабінеті платника податків" в форматі "PNG" в кодуванні Base64
   "Url": // URL посилання на чек в "Електронному кабінеті платника податків"
   "NumFiscal": // Фіскальний номер документа
   "NumLocal": // Локальний номер документа
   "OrderDateTime": // Дата і час документа
   "Offline": // Документ зареєстровано в режимі офлайн (false/true)
   "Visualization": // Документ у вибраному форматі (текст UTF-8, HTML, PDF або PNG) в кодуванні Base64
}
```

Реєстрація Z-звіту: *RegisterZRep*

Запит реєструє Z-звіт на ПРРО.

Параметри запиту:

- "Command": "RegisterZRep"
- "NumFiscal": "Фіскальний номер ПРРО"
- "ZRep": "Дані Z-звіту (null - заповнюються автоматично)"
- "ZRepToConvert": "Дані Z-звіту (XML документ у форматі НБУ в кодуванні Base64) (null - заповнюються автоматично)"
- "Visualization": "Надати візуалізацію документа" (false/true)
- "VisAsHtml": "Візуалізація у форматі HTML" (false/true)
- "VisAsPdf": "Візуалізація у форматі PDF" (false/true)
- "VisAsPng": "Візуалізація у форматі PNG" (false/true)
- "VisWidth": "Ширина області візуалізації у пікселях (0 - за замовчанням значення 440, максимальне значення 4096)"
- "VisTransparentBackground": "Прозорий фон візуалізації" (false/true)

Примітка. Параметр “ZRepToConvert” використовується у разі відсутності зазначеного параметра “ZRep”.

Примітка. Якщо жоден параметр “VisAs…” не має значення “true”, документ виводиться у текстовому UTF-8 форматі.

Приклад запиту:

```
{
  "Command": "RegisterZRep",
  "UID": "1C887BFA-3D9A-42E5-9DE5-C7BA595B1269", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "ZRep": null, // Дані Z-звіту (null - заповнюються автоматично)
  "Visualization": true // Надати документ в текстовому форматі для відображення (UTF-8) (false/true)
}
```

Параметри відповіді:

```
{
  "NumFiscal": // Фіскальний номер документа
  "NumLocal": // Локальний номер документа
  "OrderDateTime": // Дата і час документа
  "Offline": // Документ зареєстровано в режимі офлайн (false/true)
  "Visualization": // Документ у вибраному форматі (текст UTF-8, HTML, PDF або PNG) в кодуванні Base64
}
```

Запит підсумків останньої зміни: *LastShiftTotals*

Запит повертає підсумки останньої зміни на ПРРО.

Параметри та приклад запиту:

```
{
  "Command": "LastShiftTotals",
  "UID": "3E540CA4-BB75-4AC6-9CFE-F835907BFCB4", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password" // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "Visualization": true, // Надати візуалізацію документа (X-звіту) (false/true)
  "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
  "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
  "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
  "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
  "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Примітка. Якщо жоден параметр "VisAs…" не має значення "true", документ виводиться у текстовому UTF-8 форматі.

Параметри відповіді:

```
{
  "Totals": { // Підсумки зміни (якщо зміну відкрито) у форматі Z-звіту
    ...
  },
  "ShiftOpened": // Дата і час відкриття зміни (null - зміна не відкрита)
  "CashInHandPrevShift": // Залишок готівки на кінець попередньої зміни (грн)
  "CashInHand": // Залишок готівки в поточній зміні (грн)
  "Visualization": // Документ (X-звіт) у вибраному форматі (текст UTF-8, HTML, PDF або PNG) в кодуванні Base64
}
```

Примітка. Залишок готівки в поточній зміні розраховується як:

Залишок готівки на кінець попередньої зміни
+ Службове внесення/Отримання авансу/Отримання підкріплення
- Службова видача/Інкасація
- Підсумки видачі готівки
+ Підсумки реалізації. Загальна сума коштів, одержаних від клієнта ломбарду
- Підсумки реалізації. Загальна сума коштів, виданих клієнту ломбарду
+ Підсумки реалізації. Загальна сума готівкової сплати
+ -Підсумки повернення. Загальна сума коштів, одержаних від клієнта ломбарду
- -Підсумки повернення. Загальна сума коштів, виданих клієнту ломбарду
+ -Підсумки повернення. Загальна сума готівкової сплати

Встановлення залишку готівки на кінець попередньої зміни: SetCashInHandPrevShift

Запит встановлює залишок готівки для поточної відкритої або, якщо зміна не відкрита, для останньої закритої зміни.

Примітка. Запит призначений для коригування фактичного залишку готівки у разі його невідповідності фактичній сумі залишку готівки у касі. У разі коректної фіксації усіх операцій, залишок готівки обчислюється автоматично і переноситься з закритої зміни у нову відкриту зміну.

Параметри та приклад запиту:

```
{
  "Command": "SetCashInHandPrevShift",
  "UID": "5494295D-5528-4015-9FEE-5E3DDC03CAE4", / Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "CashInHandPrevShift": 123.45 // Залишок готівки на кінець попередньої зміни (грн)}
}
```

Приведення локального стану ПРРО у відповідність до стану ПРРО на Фіскальному Сервері: *Cleanup*

Якщо параметр запиту “Remove” не визначений, або має значення “false”, то:

- Запит видаляє всі локальні дані, пов’язані з ПРРО, включаючи документи, зареєстровані в режимі офлайн.
- Якщо на ПРРО відкрито зміну, то зміна закривається.
- Оновлюється збережений локально стан ПРРО.

Якщо параметр “Remove” має значення “true”, то:

- Запит видаляє всі локальні дані, пов’язані з ПРРО, включаючи документи, зареєстровані в режимі офлайн.

Параметри та приклад запиту:

```
{
   "Command": "Cleanup",
   "UID": "5494295D-5528-4015-9FEE-5E3DDC03CAE4", / Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
   "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
   "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
   "Password": "password", // Пароль секретного ключа підписувача
   "NumFiscal": 1234567890, // Фіскальний номер ПРРО
   "Remove": false, // Видалити дані ПРРО, який більше не використовується (false/true)
   "Visualization": true, // Надати візуалізацію документа (автоматично створеного Z-звіту) (false/true)
   "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
   "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
   "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
   "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
   "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Примітка. Якщо жоден параметр “VisAs…” не має значення “true”, документ виводиться у текстовому UTF-8 форматі.

Параметри відповіді:

```
{
   "ZRepAutoInfo": { // Відомості автоматично створеного Z-звіту
      …
   },
   "CloseShiftAutoInfo": { // Відомості автоматично створеного документу на закриття зміни
      …
   },
   "Visualization": // Документ у вибраному форматі (текст UTF-8, HTML, PDF або PNG) в кодуванні Base64
}
```

Запит чека з сервера: *GetCheck*

Запит повертає фіскальний чек з фіскального сервера у форматі, зазначеному параметром "Type".

Параметри запиту:

- "Command": "GetCheck"
- "RegistrarNumFiscal": "Фіскальний номер ПРРО"
- "NumFiscal": "Фіскальний номер документа"
- "Type": "Тип даних запита документа"
- "GetQrCode": "Одержати QR-код посилання на чек в "Електронному кабінеті платника податків"" (false/true),
- "QrSize": "Бажаний приблизний розмір QR-коду по вертикалі/горизонталі в пікселях (від 256 до 2048, за замовчуванням 512)"
- "Visualization": "Надати візуалізацію документа" (false/true)
- "VisAsHtml": "Візуалізація у форматі HTML" (false/true)
- "VisAsPdf": "Візуалізація у форматі PDF" (false/true)
- "VisAsPng": "Візуалізація у форматі PNG" (false/true)
- "VisWidth": "Ширина області візуалізації у пікселях (0 - за замовчанням значення 440, максимальне значення 4096)"
- "VisTransparentBackground": "Прозорий фон візуалізації" (false/true)

Приклад запиту:

```
{
  "Command": "GetCheck",
  "RegistrarNumFiscal": 1234567890, // Фіскальний номер ПРРО
  "NumFiscal": "111111", // Фіскальний номер документа
  "Type": "Visualization", // Тип даних запита документа
  "GetQrCode": true // Одержати QR-код посилання на чек в "Електронному кабінеті платника податків" (false/true)
}
```

Параметри відповіді:

```
{
  "Data": // Результат запиту або візуалізація у вибраному форматі в кодуванні Base64
  "QrCode": // QR-код посилання на чек в "Електронному кабінеті платника податків" в форматі "PNG" в кодуванні Base64
  "Url": // URL посилання на чек в "Електронному кабінеті платника податків"
  "ResultCode": // Код результату обробки запиту документа
  "ResultText": // Опис результату обробки запиту документа
}
```

Запит Z-звіту з сервера: *GetZRep*

Запит повертає Z-звіт з фіскального сервера у форматі, зазначеному параметром "Type".

Параметри та приклад запиту:

```
{
  "Command": "GetZRep",
  "UID": "4F6B19C9-C4FE-42FE-9F41-5F33A23C252A", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "RegistrarNumFiscal": 1234567890, // Фіскальний номер ПРРО
  "NumFiscal": "123456", // Фіскальний номер документа
  "Type": "Visualization", // Тип даних запиту документа
  "Visualization": true, // Надати візуалізацію документа (false/true)
  "VisAsHtml": false, // Візуалізація у форматі HTML (false/true)
  "VisAsPdf": false, // Візуалізація у форматі PDF (false/true)
  "VisAsPng": false, // Візуалізація у форматі PNG (false/true)
  "VisWidth": 0, // Ширина області візуалізації у пікселях (0 - за замовчанням значення 440,максимальне значення 4096)
  "VisTransparentBackground": false // Прозорий фон візуалізації (false/true)
}
```

Параметри відповіді:

```
{
  "Data": // Результат запиту або візуалізація у вибраному форматі в кодуванні Base64
  "ResultCode": // Код результату обробки запиту документа
  "ResultText": // Опис результату обробки запиту документа
}
```

Запит переліку змін за період: *Shifts*

Запит повертає перелік змін ПРРО за період.

Параметри запиту:

- "Command": "Shifts"
- "NumFiscal": "Фіскальний номер ПРРО"
- "From": "Дата і час початку періоду"
- "To": "Дата і час завершення періоду"

Дата і час представлені текстом у форматі ISO 8601 (наприклад, "2018-10-17T01:23:00+03:00")

Приклад запиту:

```
{
  "Command": "Shifts",
  "UID": "505F839D-E503-4CF7-94F1-0B3F7EB90B9D", // Не обов'язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
```

```
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "From": "2019-10-17T01:23:00+03:00", // Дата і час початку періоду
  "To": "2021-03-01T00:00:00+03:00" // Дата і час завершення періоду
}
```

Приклад відповіді:

```
{
  "Shifts": [ // Масив змін за період
    {
      "ShiftId": // Ідентифікатор зміни
      "OpenShiftFiscalNum": // Фіскальний номер документа "Відкриття зміни"
      "CloseShiftFiscalNum": // Фіскальний номер документа “Закриття зміни”
      "Testing": // Ознака зміни, що містить тестові документи (false/true)
      "Opened": // Дата і час відкриття зміни
      "OpenName": // П.І.Б. оператора, що відкрив зміну
      "OpenSubjectKeyId": // Ідентифікатор ключа суб’єкта сертифіката оператора
      "Closed": // Дата і час закриття зміни
      "CloseName": // П.І.Б. оператора, що закрив зміну
      "CloseSubjectKeyId": // Ідентифікатор ключа суб’єкта сертифіката оператора
      "ZRepFiscalNum": // Фіскальний номер Z-звіту
    },
    {
      …
    }
  ]
}
```

Запит переліку документів зміни: *Documents*

Запит повертає перелік документів зазначеної зміни.

Параметри та приклад запиту:

```
{
  "Command": "Documents",
  "UID": "831E23D7-A956-450F-A718-58DF9CFE60BA", // Не обов’язковий унікальний
ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "ShiftId": 12345, // Ідентифікатор зміни
  "OpenShiftFiscalNum": "123456" // Фіскальний номер документа "Відкриття зміни"
}
```

Параметри відповіді:

```
{
  "Documents": [ // Масив документів
    {
      "NumFiscal": // Фіскальний номер документа
      "NumLocal": // Локальний номер документа
      "DocDateTime": // Дата і час операції, зафіксованої документом
      "DocClass": // Клас документа (“Check”, “ZRep”)
      "CheckDocType": // Тип чека (“SaleGoods”, …)
      "CheckDocSubType": // Розширений тип чека (“CheckGoods”, …)
      "Revoked": // Ознака відкликаного документа
      "Storned": // Ознака сторнованого документа
    },
    {
      …
    }
  ]
}
```

Запит відомостей про документ за локальним номером: *DocumentInfoByLocalNum*

Запит повертає інформацію по документу за локальним номером.

Параметри та приклад запиту:

```
{
  "Command": "DocumentInfoByLocalNum",
  "UID": "0C393B51-4934-49B3-A293-642B9D5164F7", // Не обов’язковий унікальний ідентифікатор запиту, аналогічний значенню GUID (null-не використовується)
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "NumLocal": 1 // Локальний номер документа
}
```

Параметри відповіді:

```
{
  "NumFiscal": // Фіскальний номер документа
  "DocClass": // Клас документа ("Check", "ZRep")
  "CheckDocType": // Тип чека ("SaleGoods", …)
  "CheckDocSubType": // Розширений тип чека ("CheckGoods", …)
  "Revoked": // Ознака відкликаного документа
  "Storned": // Ознака сторнованого документа
}
```

Налаштування ПРРО: *SetupRegistrar*

Запит встановлює значення параметрів налаштування ПРРО.

Параметри та приклад запиту:

```
{
  "Command": "SetupRegistrar",
  "NumFiscal": 1234567890, // Фіскальний номер ПРРО
  "WorkMode": "Normal", // Режим роботи ПРРО
  "SendToCabinet": true, // Надсилати документи до кабінету користувача Cashalot (false/true)
  "AutoCloseShift": "00:30", // Час автоматичного закриття зміни (ГГ:ХХ) (null – не використовується)
  "AutoServiceDeposit": true, // Автоматична реєстрація службового внесення коштів (у національній валюті) при відкритті зміни
  "AutoServiceDepositSum": 100.00, // Сума автоматичного службового внесення коштів (у національній валюті) при відкритті зміни
  "AutoServiceIssue": true, // Автоматична реєстрація службової видачі коштів (у національній валюті) при закритті зміни (false/true)
  "ShowCashier": false, // При створенні візуалізації чеків відображати ім'я касира (false/true)
  "OmitLogos": false, // При створенні візуалізації документів не відображати логотипи (false/true)
  "ShowCashInHandOpening": false, // При створенні візуалізації Z-звітів та X-звітів відображати вхідний залишок готівки в касі (false/true)
  "ShowCashInHandCurrent": false, // При створенні візуалізації Z-звітів та X-звітів відображати поточний залишок готівки в касі (false/true)
  "ShowZeroCashInHand": false, // Відображати нульовий залишок готівки в касі (false/true)
  "PreventExceedingCashInHand": false, // При реєстрації чеків не дозволяти перевищення залишку готівки в касі (false/true)
  "ZeroingCashInHandOnShiftOpening": false, // Виконувати обнулення залишку готівки в касі на кінець попередньої зміни при відкритті зміни (false/true)
  "AutoServiceDepositCashInHand": false, // Автоматична реєстрація службового внесення коштів (у національній валюті) при відкритті зміни на суму залишку готівки в касі на кінець попередньої зміни та обнулення залишку готівки в касі на кінець попередньої зміни (false/true)
  "BlackAndWhite": true, // При створенні візуалізації документів у форматі HTML використовувати чорно-білий (true) або кольоровий (false) стандартний логотип Cashalot та колір тексту
  "VisFormatVersion": 1, // Версія формату візуалізації документів (0, 1)
  "LogoHtmlTag": null, // Елемент HTML розмітки користувацького логотипа ПРРО в кодуванні Base64 (null - не використовується) (Використовується для “VisFormatVersion” = 0)
  "UpperLogoHtmlTag ": null, // Елемент HTML розмітки верхнього користувацького логотипа ПРРО в кодуванні Base64 (null - використовується стандартний логотип Cashalot) (Використовується для “VisFormatVersion” = 1)
  "LowerLogoHtmlTag ": null // Елемент HTML розмітки нижнього користувацького логотипа ПРРО в кодуванні Base64 (null - використовується стандартний логотип
```

```
Cashalot) (Використовується для “VisFormatVersion” = 1)
}
```

Примітка 1. До кабінету користувача Cashalot надсилаються документи, що зареєстровані на ФСКО.

Примітка 2. Автоматична реєстрація службової видачі здійснюється на повну суму готівкових коштів станом на завершення зміни. Розрахунок суми службової видачі здійснюється як

Сума =
Залишок готівки на кінець попередньої зміни
+ Службове внесення/Отримання авансу/Отримання підкріплення
- Службова видача/Інкасація
- Підсумки видачі готівки
+ Підсумки реалізації. Загальна сума коштів, одержаних від клієнта ломбарду
- Підсумки реалізації. Загальна сума коштів, виданих клієнту ломбарду
+ Підсумки реалізації. Загальна сума готівкової сплати
+ -Підсумки повернення. Загальна сума коштів, одержаних від клієнта ломбарду
- -Підсумки повернення. Загальна сума коштів, виданих клієнту ломбарду
+ -Підсумки повернення. Загальна сума готівкової сплати

Примітка 3. Якщо параметр “AutoServiceDepositCashInHand” має значення true, то параметри “AutoServiceDeposit”, “AutoServiceDepositSum” та “ZeroingCashInHandOnShiftOpening” ігноруються.

Примітка 4. Приклад значення параметра “LogoHtmlTag”, “UpperLogoHtmlTag”, “LowerLogoHtmlTag” (перед кодуванням Base64):

Запит налаштувань ПРРО: *RegistrarSettings*

Параметри та приклад запиту:

```
{
  "Command": "RegistrarSettings",
  "NumFiscal": 1234567890 // Фіскальний номер ПРРО
}
```

Параметри відповіді:

```
{
  "WorkMode": // Режим роботи ПРРО
  "SendToCabinet": // Надсилати документи до кабінету користувача (false/true)
  "AutoCloseShift": // Час автоматичного закриття зміни (ГГ:ХХ) (null – не використовується)
  "AutoServiceDeposit": // Автоматична реєстрація службового внесення коштів (у національній валюті) при відкритті зміни
  "AutoServiceDepositSum": // Сума автоматичного службового внесення коштів (у національній валюті) при відкритті зміни
```

```
  "AutoServiceIssue": // Автоматична реєстрація службової видачі коштів (у національній валюті) при закритті зміни (false/true)
  "ShowCashier": // При створенні візуалізації чеків відображати ім'я касира (false/true)
  "OmitLogos": // При створенні візуалізації документів не відображати логотипи (false/true)
  "ShowCashInHandOpening": // При створенні візуалізації Z-звітів та X-звітів відображати вхідний залишок готівки в касі (false/true)
  "ShowCashInHandCurrent": // При створенні візуалізації Z-звітів та X-звітів відображати поточний залишок готівки в касі (false/true)
  "ShowZeroCashInHand": // Відображати нульовий залишок готівки в касі (false/true)
  "PreventExceedingCashInHand": // При реєстрації чеків не дозволяти перевищення залишку готівки в касі (false/true)
  "ZeroingCashInHandOnShiftOpening": // Виконувати обнулення залишку готівки в касі на кінець попередньої зміни при відкритті зміни (false/true)
  "AutoServiceDepositCashInHand": // Автоматична реєстрація службового внесення коштів (у національній валюті) при відкритті зміни на суму залишку готівки в касі на кінець попередньої зміни та обнулення залишку готівки в касі на кінець попередньої зміни (false/true)
  "BlackAndWhite": // При створенні візуалізації документів у форматі HTML використовувати чорно-білий (true) або кольоровий (false) стандартний логотип Cashalot та колір тексту
  "VisFormatVersion": // Версія формату візуалізації документів (0, 1)
  "LogoHtmlTag": // Елемент HTML розмітки користувацького логотипа ПРРО в кодуванні Base64 (null - не використовується) (Використовується для “VisFormatVersion” = 0)
  "UpperLogoHtmlTag ": // Елемент HTML розмітки верхнього користувацького логотипа ПРРО в кодуванні Base64 (null - використовується стандартний логотип Cashalot) (Використовується для “VisFormatVersion” = 1)
  "LowerLogoHtmlTag ": // Елемент HTML розмітки нижнього користувацького логотипа ПРРО в кодуванні Base64 (null - використовується стандартний логотип Cashalot) (Використовується для “VisFormatVersion” = 1)
}
```

Запит інформації сертифіката: *CertificateInfo*

Параметри та приклад запиту:

```
{
  "Command": "CertificateInfo",
  "Certificate": "cert" // Сертифікат підписувача в кодуванні Base64
}
```

Параметри відповіді:

```
{
  "SubjectKeyId": // Ідентифікатор ключа суб'єкта,
  "SerialNumber": // Серійний номер сертифіката,
  "Issuer": // Найменування видавця,
  "IssuerId": // Ідентифікатор видавця,
  "DtBeg": // Дата початку дії сертифіката,
  "DtEnd": // Дата закінчення дії сертифіката,
```

```
  "Edrpou": // ЄДРПОУ установи,
  "Drfo": // ДРФО особи,
  "Name": // ПІБ особи або найменування установи,
  "Stamp": // Сертифікат є Електронною печаткою,
  "ForSign": // Дата початку дії сертифіката,
  "ForEncrypt": // Дата початку дії сертифіката
}
```

Отримання переліку сертифікатів з контейнера у форматі PFX/JKS: *ContainerCertificates*

Параметри та приклад запиту:

```
{
  "Command": "ContainerCertificates",
  "PrivateKey": "container" // Контейнер PFX/JKS в кодуванні Base64,
  "Password": "password" // Пароль секретного ключа підписувача
}
```

Параметри відповіді:

```
{
  "Certificates": [ // Відомості про сертифікати
    {
      "Certificate": // Сертифікат в кодуванні Base64,
      "SubjectKeyId": // Ідентифікатор ключа суб'єкта,
      "SerialNumber": // Серійний номер сертифіката,
      "Issuer": // Найменування видавця,
      "IssuerId": // Ідентифікатор видавця,
      "DtBeg": // Дата початку дії сертифіката,
      "DtEnd": // Дата закінчення дії сертифіката,
      "Edrpou": // ЄДРПОУ установи,
      "Drfo": // ДРФО особи,
      "Name": // ПІБ особи або найменування установи,
      "Stamp": // Сертифікат є Електронною печаткою,
      "ForSign": // Дата початку дії сертифіката,
      "ForEncrypt": // Дата початку дії сертифіката
    }
  ]
}
```

Конвертування XML документа у форматі НБУ: *ConvertDocument*

Параметри запиту:

- "Command": "ConvertDocument"
- "NumFiscal": "Фіскальний номер ПРРО"
- "Doc": "XML документ у форматі НБУ в кодуванні Base64"

Параметри відповіді:

- "Check": "Дані фіскального чека"

- "ZRep": "Дані Z-звіту"

Надсилання чека споживачу: *SendCheckToConsumer*

Параметри та приклад запиту:

```
{
  "Command": "SendCheckToConsumer",
  "Certificate": "cert", // Сертифікат підписувача в кодуванні Base64
  "PrivateKey": "key", // Секретний ключ підписувача в кодуванні Base64
  "Password": "password", // Пароль секретного ключа підписувача
  "ServiceType": 0, // Сервіс надсилання повідомлення
  "PhoneNumber": "+380631111111", // Номер телефону споживача
  "RegistrarNumFiscal": 4000027822, // Фіскальний номер ПРРО
  "OrderNumFiscal": "10022342345345", // Фіскальний номер документа
  "OrderSum": 123.45, // Сума документа
  "OrderDateTime: "2025-09-19T12:45" // Дата і час документа
}
```

Примітка. Для використання команди, необхідно налаштувати параметри провайдера розсилки повідомлень у кабінеті користувача Cashalot.

Структури даних

Фіскальний чек

Використовується в запиті «RegisterCheck»

```
public class CheckContent {
    public CHead CHECKHEAD { get; set; }
    public CTotal CHECKTOTAL { get; set; }
    public CPayRow[] CHECKPAY { get; set; }
    public CTaxRow[] CHECKTAX { get; set; }
    public CPtks CHECKPTKS { get; set; }
    public CDutyFreeRow[] CHECKDUTYFREE { get; set; }
    public CDutyFreeRemainsRow[] CHECKDUTYFREEREMAINS { get; set; }
    public CBodyRow[] CHECKBODY { get; set; }
}

public class CHead {
    public CheckDocumentType DOCTYPE { get; set; }
    public CheckDocumentSubType DOCSUBTYPE { get; set; }
    public string UID { get; set; }
    public string TIN { get; set; }
    public string IPN { get; set; }
    public string ORGNM { get; set; }
    public string IPNBUYER { get; set; }
    public string POINTNM { get; set; }
    public string POINTADDR { get; set; }
    public string ORDERDATE { get; set; }
    public string ORDERTIME { get; set; }
    public string ORDERNUM { get; set; }
    public string CASHDESKNUM { get; set; }
    public string CASHREGISTERNUM { get; set; }
    public string ORDERRETNUM { get; set; }
    public string ORDERSTORNUM { get; set; }
    public string OPERTYPENM { get; set; }
    public string VEHICLERN { get; set; }
    public bool REVOKELASTONLINEDOC { get; set; }
    public string CASHIER { get; set; }
    public string LOGOURL { get; set; }
    public string COMMENT { get; set; }
    public string VER { get; set; }
    public string ORDERTAXNUM { get; set; }
    public bool REVOKED { get; set; }
    public bool STORNED { get; set; }
    public bool TESTING { get; set; }
    public bool OFFLINE { get; set; }
    public string PREVDOCHASH { get; set; }
}

public class CExciseLabelsRow {
    public string EXCISELABEL { get; set; }
}
```

```
public class CBodyRow {
    public string CODE { get; set; }
    public string BARCODE { get; set; }
    public string UKTZED { get; set; }
    public string DKPP { get; set; }
    public string NAME { get; set; }
    public string DESCRIPTION { get; set; }
    public int UNITCD { get; set; }
    public string UNITNM { get; set; }
    public decimal AMOUNT { get; set; }
    public decimal TOBACCOWEIGHT { get; set; }
    public int TOBACCOQT { get; set; }
    public decimal ALCOSTRENGTH { get; set; }
    public decimal ALCOVOL { get; set; }
    public decimal PRICE { get; set; }
    public string LETTERS { get; set; }
    public decimal COST { get; set; }
    public string RECIPIENTNM { get; set; }
    public string RECIPIENTIPN { get; set; }
    public string BANKCD { get; set; }
    public string BANKNM { get; set; }
    public string BANKRS { get; set; }
    public string PAYDETAILS { get; set; }
    public string PAYPURPOSE { get; set; }
    public string PAYERNM { get; set; }
    public string PAYERIPN { get; set; }
    public string PAYERPACTNUM { get; set; }
    public string PAYDETAILSP { get; set; }
    public string BASISPAY { get; set; }
    public string PTKSDEPTNUM { get; set; }
    public string PROVIDERNM { get; set; }
    public string PROVIDERTIN { get; set; }
    public string PROVIDERRCPRS { get; set; }
    public string TRANSFERID { get; set; }
    public string PAYOUTTYPE { get; set; }
    public string FUELORDERNUM { get; set; }
    public string FUELUNITNM { get; set; }
    public string FUELTANKNUM { get; set; }
    public string FUELCOLUMNNUM { get; set; }
    public string FUELFAUCETNUM { get; set; }
    public bool FUELSALESIGN { get; set; }
    public int VALCD { get; set; }
    public string VALSYMCD { get; set; }
    public string VALNM { get; set; }
    public int? VALOPERTYPE { get; set; }
    public string VALOUTCD { get; set; }
    public string VALOUTSYMCD { get; set; }
    public string VALOUTNM { get; set; }
    public decima VALCOURSE { get; set; }
    public string VALCOURSEDATE { get; set; }
```

```
    public decimal VALFOREIGNSUM { get; set; }
    public decimal VALNATIONALSUM { get; set; }
    public decimal VALCOMMISSION { get; set; }
    public int VALOPERCNT { get; set; }
    public bool VALREFUSESELL { get; set; }
    public bool PWNDIR { get; set; }
    public int? USAGETYPE { get; set; }
    public int? DISCOUNTTYPE { get; set; }
    public decimal SUBTOTAL { get; set; }
    public int? DISCOUNTNUM { get; set; }
    public string DISCOUNTTAX { get; set; }
    public decimal DISCOUNTPERCENT { get; set; }
    public decimal DISCOUNTSUM { get; set; }
    public decimal PARTPAYSUM { get; set; }
    public string COMMENT { get; set; }
    public CExciseLabelsRow[] EXCISELABELS { get; set; }
}

public class CPtks {
    public string PTKSPN { get; set; }
    public string PTKSNM { get; set; }
    public string PAYSYSTEMPN { get; set; }
    public string PAYSYSTEMNM { get; set; }
    public string ACQUIREPN { get; set; }
    public string ACQUIRENM { get; set; }
    public string ACQUIRETRANSID { get; set; }
    public string POSTRANSDATE { get; set; }
    public string POSTRANSNUM { get; set; }
    public string DEVICEID { get; set; }
    public string EPZDETAILS { get; set; }
    public string AUTHCD { get; set; }
    public string TERMINALNUM { get; set; }
    public string TERMINALADDR { get; set; }
    public string OPERNUM { get; set; }
    public string OPERSYSNUM { get; set; }
    public string BANKCD { get; set; }
    public string BANKNM { get; set; }
}

public class CTaxRow {
    public int TYPE { get; set; }
    public string NAME { get; set; }
    public string LETTER { get; set; }
    public decimal PRC { get; set; }
    public bool SIGN { get; set; }
    public decimal TURNOVER { get; set; }
    public decimal TURNOVERDISCOUNT { get; set; }
    public decimal SOURCESUM { get; set; }
    public decimal SUM { get; set; }
}
```

```
public class CPayRow {
    public int PAYFORMCD { get; set; }
    public string PAYFORMNM { get; set; }
    public decimal SUM { get; set; }
    public decimal PROVIDED { get; set; }
    public decimal REMAINS { get; set; }
    public CPaySysRow[] PAYSYS { get; set; }
}

public class CPaySysRow {
    public string TAXNUM { get; set; }
    public string NAME { get; set; }
    public string ACQUIREPN { get; set; }
    public string ACQUIRENM { get; set; }
    public string ACQUIRETRANSID { get; set; }
    public string POSTRANSDATE { get; set; }
    public string POSTRANSNUM { get; set; }
    public string DEVICEID { get; set; }
    public string EPZDETAILS { get; set; }
    public string AUTHCD { get; set; }
    public string CLIENTINFO { get; set; }
    public decimal SUM { get; set; }
    public decimal COMMISSION { get; set; }
}

public class CDutyFreeRow {
    public string PASSPORTDOC { get; set; }
    public string PASSPORTCOUNTRY { get; set; }
    public int VAL_CD { get; set; }
    public string VAL_SYMCD { get; set; }
    public string VAL_NM { get; set; }
    public decimal VAL_COURSE { get; set; }
    public decimal VAL_SUM { get; set; }
    public decimal VAL_PROVIDED { get; set; }
}

public class CDutyFreeRemainsRow {
    public int VAL_CD { get; set; }
    public string VAL_SYMCD { get; set; }
    public string VAL_NM { get; set; }
    public decimal VAL_COURSE { get; set; }
    public decimal VAL_SUM { get; set; }
    public decimal VAL_NATIONALSUM { get; set; }
}

public class CTotal {
    public decimal SUM { get; set; }
    public decimal PWNSUMISSUED { get; set; }
    public decimal PWNSUMRECEIVED { get; set; }
    public decimal RNDSUM { get; set; }
    public decimal NORNDSUM { get; set; }
```

```
    public decimal NOTAXSUM { get; set; }
    public decimal COMMISSION { get; set; }
    public int? USAGETYPE { get; set; }
    public decimal SUBCHECK { get; set; }
    public int? DISCOUNTTYPE { get; set; }
    public decimal DISCOUNTPERCENT { get; set; }
    public decimal DISCOUNTSUM { get; set; }
    public PartialPaymentType? PARTPAYTYPE { get; set; }
    public decimal PARTPAYPERCENT { get; set; }
    public decimal PARTPAYSUM { get; set; }
    public string PARTPAYORDPREPAYNUM { get; set; }
}
```

Z-звіт

Використовується в запитах "RegisterZRep", "LastShiftTotals"

```
public class ZRepContent {
    public ZHead ZREPHEAD { get; set; }
    public ZPay ZREPREALIZ { get; set; }
    public ZPay ZREPRETURN { get; set; }
    public ZCash ZREPCASH { get; set; }
    public ZVal ZREPVAL { get; set; }
    public ZFuel ZREPFUEL { get; set; }
    public ZBody ZREPBODY { get; set; }
}

public class ZHead {
    public string UID { get; set; }
    public string TIN { get; set; }
    public string IPN { get; set; }
    public string ORGNM { get; set; }
    public string POINTNM { get; set; }
    public string POINTADDR { get; set; }
    public string ORDERDATE { get; set; }
    public string ORDERTIME { get; set; }
    public string ORDERNUM { get; set; }
    public string CASHDESKNUM { get; set; }
    public string CASHREGISTERNUM { get; set; }
    public string CASHIER { get; set; }
    public string VER { get; set; }
    public string ORDERTAXNUM { get; set; }
    public bool REVOKED { get; set; }
    public bool TESTING { get; set; }
    public bool OFFLINE { get; set; }
    public string PREVDOCHASH { get; set; }
}

public class ZBody {
    public decimal SERVICEINPUT { get; set; }
    public decimal SERVICEOUTPUT { get; set; }
    public string VEHICLERN { get; set; }
```

```
}

public class ZFuelRemainsRow {
    public string NAME { get; set; }
    public string TANKNUM { get; set; }
    public string COLUMNNUM { get; set; }
    public string FAUCETNUM { get; set; }
    public decimal REMAINS { get; set; }
}

public class ZFuelPayFormsRow {
    public int PAYFORMCD { get; set; }
    public string PAYFORMNM { get; set; }
    public decimal SUM { get; set; }
    public decimal AMOUNT { get; set; }
}

public class ZFuelDetailsRow {
    public string NAME { get; set; }
    public decimal ACCEPTED { get; set; }
    public decimal REALIZVOL { get; set; }
    public decimal REALIZPRK { get; set; }
    public decimal REALIZCOST { get; set; }
    public decimal REMAINS { get; set; }
    public ZFuelPayFormsRow[] PAYFORMS { get; set; }
}

public class ZFuel {
    public ZFuelDetailsRow[] DETAILS { get; set; }
    public ZFuelRemainsRow[] REMAINS { get; set; }
}

public class ZValDetailsRow {
    public int VALNUM { get; set; }
    public int VALCD { get; set; }
    public string VALSYMCD { get; set; }
    public string VALNM { get; set; }
    public decimal COURSEBUY { get; set; }
    public decimal COURSESELL { get; set; }
    public decimal COURSEREG { get; set; }
    public decimal BUYVALI { get; set; }
    public decimal SELLVALI { get; set; }
    public decimal BUYVALN { get; set; }
    public decimal SELLVALN { get; set; }
    public decimal STORBUYVALI { get; set; }
    public decimal STORSELLVALI { get; set; }
    public decimal STORBUYVALN { get; set; }
    public decimal STORSELLVALN { get; set; }
    public decimal CINVALI { get; set; }
    public decimal COUTVALI { get; set; }
    public decimal UNUSVALI { get; set; }
    public decimal UNUSVALN { get; set; }
```

```
    public decimal COMMISSION { get; set; }
    public decimal INADVANCE { get; set; }
    public decimal INATTACH { get; set; }
    public decimal SURRCOLLECTION { get; set; }
    public decimal STORUNUSVALI { get; set; }
    public decimal STORUNUSVALN { get; set; }
    public decimal STORCINVALI { get; set; }
    public decimal STORCOUTVALI { get; set; }
    public decimal STORCOMMISSION { get; set; }
    public int VALCRCD { get; set; }
    public string CROSSSYMCD { get; set; }
    public decimal CROSSCOURSE { get; set; }
}

public class ZVal {
    public decimal TOTALINADVANCE { get; set; }
    public decimal TOTALINATTACH { get; set; }
    public decimal TOTALSURRCOLLECTION { get; set; }
    public decimal COMMISSION { get; set; }
    public int CALCDOCSCNT { get; set; }
    public decimal ACCEPTEDN { get; set; }
    public decimal ISSUEDN { get; set; }
    public decimal COMMISSIONN { get; set; }
    public int TRANSFERSCNT { get; set; }
    public ZValDetailsRow[] DETAILS { get; set; }
    public ZTaxRow[] TAXES { get; set; }
}

public class ZTaxRow {
    public int TYPE { get; set; }
    public string NAME { get; set; }
    public string LETTER { get; set; }
    public decimal PRC { get; set; }
    public bool SIGN { get; set; }
    public decimal TURNOVER { get; set; }
    public decimal TURNOVERDISCOUNT { get; set; }
    public decimal SOURCESUM { get; set; }
    public decimal SUM { get; set; }
}

public class ZCash {
    public decimal SUM { get; set; }
    public decimal COMMISSION { get; set; }
    public int ORDERSCNT { get; set; }
}

public class ZPayFormsRow {
    public int PAYFORMCD { get; set; }
    public string PAYFORMNM { get; set; }
    public decimal SUM { get; set; }
}
```

```
public class ZPay {
    public decimal SUM { get; set; }
    public decimal PWNSUMISSUED { get; set; }
    public decimal PWNSUMRECEIVED { get; set; }
    public decimal RNDSUM { get; set; }
    public decimal NORNDSUM { get; set; }
    public int ORDERSCNT { get; set; }
    public int PWNORDERSCNTISSUED { get; set; }
    public int PWNORDERSCNTRECEIVED { get; set; }
    public int TOTALCURRENCYCOST { get; set; }
    public decimal TOTALCURRENCYSUM { get; set; }
    public decimal TOTALCURRENCYCOMMISSION { get; set; }
    public ZPayFormsRow[] PAYFORMS { get; set; }
    public ZTaxRow[] TAXES { get; set; }
}
```

Константи

Клас документа

```
/// <summary>
///    Клас документа
/// </summary>
public enum DocumentClass {
    /// <summary>
    ///    Чек
    /// </summary>
    Check = 0,

    /// <summary>
    ///    Z-звіт
    /// </summary>
    ZRep = 1
}
```

Чек. Тип документа

```
/// <summary>
///    Чек. Тип документа
/// </summary>
public enum CheckDocumentType {
    /// <summary>
    ///    Чек реалізації товарів/послуг
    /// </summary>
    SaleGoods = 0,

    /// <summary>
    ///    Чек переказу коштів
    /// </summary>
    TransferFunds = 1,

    /// <summary>
    ///    Чек операції обміну валюти
    /// </summary>
```

```
    CurrencyExchange = 2,

    /// <summary>
    ///     Чек видачі готівки
    /// </summary>
    CashWithdrawal = 3,

    /// <summary>
    ///     Чек обслуговування у ломбарді
    /// </summary>
    Pawnshop = 4,

    /// <summary>
    ///     Відкриття зміни
    /// </summary>
    OpenShift = 100,

    /// <summary>
    ///     Закриття зміни
    /// </summary>
    CloseShift = 101,

    /// <summary>
    ///     Початок офлайн сесії
    /// </summary>
    OfflineBegin = 102,

    /// <summary>
    ///     Завершення офлайн сесії
    /// </summary>
    OfflineEnd = 103
}
```

Чек. Розширений тип документа

```
/// <summary>
///     Чек. Розширений тип документа
/// </summary>
public enum CheckDocumentSubType {
    /// <summary>
    ///     Касовий чек (реалізація)
    /// </summary>
    CheckGoods = 0,

    /// <summary>
    ///     Видатковий чек (повернення)
    /// </summary>
    CheckReturn = 1,

    /// <summary>
    ///     Чек операції «службове внесення»/«отримання авансу»
    /// </summary>
    ServiceDeposit = 2,

    /// <summary>
```

```
    ///     Чек операції «отримання підкріплення»
    /// </summary>
    AdditionalDeposit = 3,

    /// <summary>
    ///     Чек операції «службова видача»/«інкасація»
    /// </summary>
    ServiceIssue = 4,

    /// <summary>
    ///     Чек сторнування попереднього чека
    /// </summary>
    CheckStorno = 5
}
```

Чек. Тип часткової сплати

```
/// <summary>
///     Тип часткової сплати
/// </summary>
public enum PartialPaymentType {
    /// <summary>
    ///     Передплата
    /// </summary>
    Prepay = 1,

    /// <summary>
    ///     Остаточний розрахунок
    /// </summary>
    FinalPayment = 2,

    /// <summary>
    ///     Чергова сплата
    /// </summary>
    NextPayment = 3
}
```

Тип даних запита документа

```
/// <summary>
///     Тип даних запита документа
/// </summary>
public enum DocumentRequestType {
    /// <summary>
    ///     Перевірка наявності документа
    /// </summary>
    Availability = 0,

    /// <summary>
    ///     Оригінальний XML
    /// </summary>
    OriginalXml = 1,

    /// <summary>
    ///     XML засвідчений КЕП Фіскального Сервера
```

```
    /// </summary>
    SignedByServerXml = 2,

    /// <summary>
    ///     Документ в текстовому форматі для відображення (UTF-8)
    /// </summary>
    Visualization = 3,

    /// <summary>
    ///     XML засвідчений КЕП відправника
    /// </summary>
    SignedBySenderXml = 4,

    /// <summary>
    ///     XML засвідчений КЕП відправника і КЕП Фіскального Сервера
    /// </summary>
    SignedBySenderAndServerXml = 5
}
```

Режим роботи ПРРО

```
/// <summary>
///     Режим роботи ПРРО
/// </summary>
public enum RegistrarWorkMode {
    /// <summary>
    ///     Звичайний
    /// </summary>
    Normal = 0,

    /// <summary>
    ///     Режим онлайн заборонений
    /// </summary>
    DisableOnline = 1,

    /// <summary>
    ///     Режим офлайн заборонений
    /// </summary>
    DisableOffline = 2
}
```

Код результату обробки запиту документа

```
/// <summary>
///     Код результату обробки запиту документа
/// </summary>
public enum DocumentRequestResultCode {
    /// <summary>
    ///     OK
    /// </summary>
    Ok = 0,

    /// <summary>
    ///         Документ з фіскальним номером, що відповідає режиму онлайн, не зареєстрований на ПРРО
```

```
    /// </summary>
    OnlineDocumentAbsent = 1,

    /// <summary>
    ///     Фіскальний номер зарезервований для використання в режимі офлайн на ПРРО, але наразі ще не переданий
    ///     з ПРРО до контролюючого органу
    /// </summary>
    OfflineNumberReserved = 2,

    /// <summary>
    ///     Фіскальний номер не зарезервований для використання в режимі офлайн на ПРРО
    /// </summary>
    OfflineNumberNotReserved = 3,

    /// <summary>
    ///     ПРРО не зареєстрований
    /// </summary>
    TransactionsRegistrarNotRegistered = 4
}
```

Сервіси надсилання повідомлень споживачу

```
/// <summary>
///     Сервіси надсилання повідомлень споживачу
/// </summary>
public enum SendServices {
    /// <summary>
    ///     SMS
    /// </summary>
    Sms = 0,

    /// <summary>
    ///     Viber
    /// </summary>
    Viber = 1

}
```

Підтримка документів у форматі НБУ

«Документи у форматі НБУ» – це документи (чеки та Z-звіти) у форматі, опис якого міститься в документі «Технологія зберігання і збору даних РРО для ДПС. Протокол передачі
інформації» Версія 2.1.9, Редакція 1.4 (https://tax.gov.ua/baneryi/programni-rro/opis-ari-fiskalnogo-servera/opis-ari--elektronniy-kabinet-/).

Документи у форматі НБУ можуть використовуватися в якості вхідних даних для команд

- Реєстрація фіскального чека
- Реєстрація Z-звіту

Команда Конвертування XML документа у форматі НБУ дозволяє отримати Фіскальний чек та Z-звіт у форматі API взаємодії з фіскальним сервером на підставі відповідних документів у форматі НБУ.

Приклади документів у форматі НБУ

Фіскальний чек

```
<?xml version="1.0" encoding="windows-1251"?>
<RQ V="1">
        <DAT FN="4000000001" TN="34554362" DI="17" ZN="0" V="1">
            <C T="0">
                <L N="1">
                </L>
                <P N="2" C="0" NM="  whirlpool mt 744()" SM="020" Q="2000" PRC="10" TX="1"/>
                <P N="3" C="0" NM=" bosch mfq 1931()" SM="010" Q="1000" PRC="10" TX="2"/>
                <P N="4" C="0" NM=" bosch mfq 1943()" SM="12600" Q="1000" PRC="12600" TX="3"/>
                <P N="5" C="0" NM=" bosch mfq 1933()" SM="12600" Q="1000" PRC="12600" TX="4"/>
                <P N="6" C="0" NM=" bosch mfq 1934()" SM="12600" Q="1000" PRC="12600" TX="5"/>
                <M N="7" T="0" NM="" SM="30"/>
                <M N="8" T="1" NM="" SM="25200"/>
                <M N="9" T="2" NM="" SM="12600"/>
                <E N="10" NO="6" SM="37830" TS="20200801114528" CS="1111111111">
                    <TX TX="1" TXPR="20" TXSM="2003" DTPR="500" DTSM="600" TXTY="0" TXAL="2"/>
                    <TX TX="2" TXPR="0" TXSM="000" DTPR="5.00" DTSM="600" TXTY="0" TXAL="2"/>
                    <TX TX="3" TXPR="7" TXSM="824" DTPR="0.00" DTSM="0" TXTY="0" TXAL="0"/>
                </E>
            </C>
            <TS>20200801114528</TS>
        </DAT>
```

<MAC>9EA16642B5BA427D5F8EF59862AA5FE9A39483448202088BC1C8CCA58D65588B</MAC>
</RQ>

Z-звіт

```
<?xml version="1.0" encoding="windows-1251"?>
<RQ V="1">
        <DAT FN="4000000001" TN="34554362" DI="20" ZN="0" V="1">
            <Z NO="3">
                <TXS N="А" TXPR="20" TXI="14000" TXO="000" SMI="140" SMO="020" DTPR="000" DTI="000" DTO="0" TXTY="0" TXAL="0"/>
                <TXS N="Б" TXPR="0" TXI="000" TXO="000" SMI="070" SMO="010" DTPR="000" DTI="000" DTO="0" TXTY="0" TXAL="0"/>
                <TXS N="В" TXPR="7" TXI="5600" TXO="000" SMI="88200" SMO="000" DTPR="000" DTI="000" DTO="0" TXTY="0" TXAL="0"/>
                <TXS N="ГА" TXPR="20" TXI="4200" TXO="000" SMI="88200" SMO="000" DTPR="000" DTI="000" DTO="0" TXTY="0" TXAL="0"/>
                <TXS N="ГБ" TXPR="0" TXI="4200" TXO="000" SMI="88200" SMO="000" DTPR="000" DTI="000" DTO="0" TXTY="0" TXAL="0"/>
                <M NM="Готівка" SMI="210" SMO="30"/>
                <M NM="Картка" SMI="176400" SMO="000"/>
                <M NM="Кредит" SMI="88200" SMO="000"/>
                <NC NI="7" NO="1"/>
                <IO NM="ГОТІВКА" SMI="000" SMO="000"/>
            </Z>
            <TS>20200801115654</TS>
        </DAT>

<MAC>2FBCB89CB20F78E42C51A64E717781D14ECAAE00B04FDF55BCEAA402A2E87733</MAC>
</RQ>
```