

Автоматизированная Информационная Система

**«Е-Заявление»**

**Техническая инструкция сервисов по Справке  
о привлечении лица к уголовной  
ответственности, о наличии либо отсутствии у  
лица судимости на территории  
Кыргызской Республики  
версия сервисов V2, в том числе справка на 3  
часа**

Сервис работает посредством СМЭВ Түндүк, путем реализации веб-сервиса согласно протоколу СМЭВ.

Для получения WSDL файла необходимо обратиться по ссылке адреса в сервер безопасности СМЭВ Түндүк.

### Терминология:

СМЭВ - система межведомственного электронного взаимодействия.

Сервер безопасности - Шлюз подключения к СМЭВ.

Веб-сервис разработан методом протокола SOAP.

### Условные обозначения:

В этом разделе описываются заголовки SOAP, которые используются СМЭВ Түндүк.

В таблице ниже представлены заголовки СМЭВ Түндүк.

Когда клиент службы отправляет запрос на сервер безопасности, обязательно должна присутствовать одна из служб полей или centralService. Если используется поле centralService, сервер безопасности разрешает центральную службу и автоматически заполняется в поле службы идентификатором конкретной услуги, которая реализует центральную службу.

Поле	Тип	Обязательное или Опциональное	Описание
client	XRoadClientIdentifierType	Обязательное	Идентифицирует Клиента — сущность, которая инициирует вызов услуги.
service	XRoadServiceIdentifierType	Опциональное	Идентифицирует сервис, вызываемый запросом.
centralService	XRoadCentralServiceIdentifierType	Опциональное	Идентифицирует центральную службу, вызываемую запросом.
id	string	Обязательное	Уникальный идентификатор для сообщения. Рекомендуемый метод генерации сообщения ID - 30 байт случайных данных, в кодировке Base64.

userId	string	Обязательное	Пользователь, действие которого инициировало запрос. Идентификатор пользователя должен быть с двухбуквенным ISO код страны (например, KG20114199609123).
requestHash	string	Оptionальное	Для ответов. Это поле содержит хэш запроса SOAP сообщения. Это поле автоматически заполняется поставщиком сервисов сервера безопасности.
requestHash/ @algorithmId	string	Обязательное	Идентифицирует хэшалгоритм, который был использован для вычисления значения запроса поля Hash. Алгоритмы обозначаются как URI перечисленные в XML-DSIG спецификации [DSI13].
issue	string	Оptionальное	Идентифицирует заявление, основание или документ, который послужил основанием обращения к сервису. Это поле используется для подключения службы запросов (и ответов) к рабочим процедурам
authentication Method	string	Обязательное	Метод аутентификации
profileVersion	string	Обязательное	Версия протокола СМЭВ Түндүк. Для СМЭВ Түндүк 6 версия протокола 4.0

При ответе служба копирует все поля заголовка из запроса в ответ.

Поле requestHash используется для создания прочной связи между запросом и ответом.

Таким образом, можно подтвердить, что определенная запись реестра возвращается в ответ на определенный запрос. Используя журналы, можно восстановить и проверить пару запрос-ответ.

Поле requestHash автоматически создается сервером безопасности поставщика услуг и проверяется сервером безопасности клиента службы.

Поле authenticationMethod должно быть одним из следующих:

EID - с удостоверением личности (ID-CARD, DIGI-ID);

MID - с мобильным удостоверением личности (MOBILE-ID);

CERT - с другим сертификатом;

EXTERNAL - через стороннюю услугу (BANK-LINK);

PASSWORD - с идентификатором пользователя и паролем;

SYSTEM - когда запрос был отправлен заданием cron или другим автоматическим процессом.

*Данные заголовки являются стандартными для использования СМЭВ Тундук и присутствуют во всех методах.*

**Описание методов сервиса и их индивидуальные параметры:***ERecordV2StatementVerifyDataByPinResponse****ERecordV2StatementVerifyDataByPin***(*ERecordV2StatementVerifyDataByPinRequest request*) - «Е-Заявление» - Проверка данных по ПИН.

А также:

*ERecordV2StatementVerifyDataByPinTestResponse****ERecordV2StatementVerifyDataByPinTest***(*ERecordV2StatementVerifyDataByPinTestRequest request request*) - «Е-Заявление» (тестовая среда) - Проверка данных по ПИН.

Где:

request – параметры запроса с заголовком СМЭВ Түндүк и дополнительные параметры;

companyIdentifier - полученный уникальный персональный идентификатор;

pin - персональный код заявителя;

statementTypeCode - тип услуги, код из реестра -

1 день 1

3 дня 2

7 дней 3 (на бесплатной основе, предназначен для гос. портала, при

запросе на 7 дней StatementTypeSumOfService - сумма услуги, возвращается 0, и не требует ERecordV2StatementConfirmPayment Подтверждение оплаты)

3 часа 4 новый модуль «3 часа»

regionCode - место получения справки, код из реестра полученный из сервиса список мест получения справок (ERecordV2ListPlacesOfReceive - место получения справок согласно типа услуги - statementTypeCode)

**Примечание:**для тестовой среды (*ERecordV2StatementVerifyDataByPinTest*) regionCode используется из следующего списка:

	regionCode
Бишкек	1
Ош	2
Каракол	3
Нарын	4
Талас	5
Баткен	6
Жалал-Абад	7

personPhone - номер телефона заявителя;

withVariablePersonalData – значение true, если заявитель указал, что справка с измененными персональными данными, если нет, то false.

Сервис возвращает:

a)

Code = 0;

Name = "Success";

Успешно, в этом случае возвращается объект ObjectResult где:

SecretKey - идентификатор подтверждения оплаты, используется в методе *ERecordV2StatementConfirmPayment* (*ERecordV2StatementConfirmPaymentTest*) для подтверждения оплаты.

DataIdentifier - идентификатор данных, используется в методе *ERecordV2StatementConfirmPayment* (*ERecordV2StatementConfirmPaymentTest*) для подтверждения оплаты.

PersonName, PersonSurname, PersonPatronymic - имя, фамилия, отчество, заявителя.

ReceiptRegionName, ReceiptRegionCode - место получения справки, наименование и код.

StatementTypeName, StatementTypeCode - тип услуги, наименование и код.

StatementTypeSumOfService - сумма услуги.

ReceivingDate, ReceivingComment – дата/время, сообщение, когда будет готова справка (поля предназначены только для модуля «3 часа»), поля для экономических агентов, которые, в свою очередь, будут отображать их в удобной для клиента форме. Клиенту в свою очередь в интерфейсе агента отобразятся правила принятия условий в виде отображения «ДА»/«НЕТ». В случае, если клиент согласен, пропускается оплата.

b)

Code = 1;

Name = "ServiceError";

Нет подключения или внутренняя ошибка!

ObjectResult = null

c)

Code = 2;

Name = "InvalidCompanyIdentifier ";

Неверный идентификатор!

ObjectResult = null

d)

```
Code = 4;  
Name = "PINIsEmpty";
```

ПИН не указан!  
ObjectResult = null

e)

```
Code = 5;  
Name = "PINMustContain14Characters";
```

ПИН должен содержать 14 цифр!  
ObjectResult = null

f)

```
Code = 6;  
Name = "PersonNotFound";
```

Не были найдены данные заявителя!  
ObjectResult = null

g)

```
Code = 7;  
Name = "RegionNotFound";
```

Указанное место получения не существует!  
ObjectResult = null

h)

```
Code = 8;  
Name = "StatementTypeNotFound";
```

Указанный тип услуги не существует!  
ObjectResult = null

j)

```
Code = x;  
Name = "Другой статус";
```

Другие возникшие статусы во время использования сервиса. В этом случае:  
ObjectResult = null

**Пример запроса:**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xro="http://x-road.eu/xsd/xroad.xsd" xmlns:iden="http://x-road.eu/xsd/identifiers"
xmlns:erec="http://erecordV2_service.x-road.ee/erecordV2">
  <soapenv:Header>
    <xro:protocolVersion>4.0</xro:protocolVersion>
    <xro:issue></xro:issue>
    <xro:id></xro:id>
    <xro:userId></xro:userId>
    <xro:service iden:objectType="SERVICE">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>GOV</iden:memberClass>
      <iden:memberCode>70000006</iden:memberCode>
      <iden:subsystemCode>record-service</iden:subsystemCode>
      <iden:serviceCode>ERecordV2StatementVerifyDataByPinTest</iden:serviceCode>
      <iden:serviceVersion>v1</iden:serviceVersion>
    </xro:service>
    <xro:client iden:objectType="SUBSYSTEM">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>YouMemberClass</iden:memberClass>
      <iden:memberCode>YouMemberCode</iden:memberCode>
      <iden:subsystemCode>YouSubsystemCode</iden:subsystemCode>
    </xro:client>
  </soapenv:Header>
  <soapenv:Body>
    <erec:ERecordV2StatementVerifyDataByPinTest>
      <erec:companyIdentifier>0000000-0000-0000-0000-
000000000</erec:companyIdentifier>
      <erec:pin>12345678912345</erec:pin>
      <erec:regionCode>1</erec:regionCode>
      <erec:statementTypeCode>4</erec:statementTypeCode>
      <erec:personPhone>443355465</erec:personPhone>
      <erec:withVariablePersonalData>true</erec:withVariablePersonalData>
    </erec:ERecordV2StatementVerifyDataByPinTest>
  </soapenv:Body>
</soapenv:Envelope>

```

*ERecordV2StatementVerifyDataByBarcodeResponse*

***ERecordV2StatementVerifyDataByBarcode***

(*ERecordV2StatementVerifyDataByBarcodeRequest request*) - «Е-Заявление» - Проверка данных по 16 значному коду.

А также:

*ERecordV2StatementVerifyDataByBarcodeTestResponse*

***ERecordV2StatementVerifyDataByBarcodeTest***

(*ERecordV2StatementVerifyDataByBarcodeTestRequest request*) - «Е-Заявление» (тестовая среда) - Проверка данных по 16 значному коду.

Где:

request – параметры запроса с заголовком СМЭВ Түндүк и дополнительные параметры;

companyIdentifier - полученный уникальный персональный идентификатор;

dataIdentifier - 16 цифр - идентификатор данных, указанный плательщиком.

Сервис возвращает:

а)

Code = 0;

Name = "Success";

Успешно, в этом случае возвращается объект ObjectResult, где:

SecretKey - идентификатор подтверждения оплаты, используется в методе *ERecordV2StatementConfirmPayment* (*ERecordV2StatementConfirmPaymentTest*) для подтверждения оплаты.

DataIdentifier - идентификатор данных, используется в методе *ERecordV2StatementConfirmPayment* (*ERecordV2StatementConfirmPaymentTest*) для подтверждения оплаты.

PersonName, PersonSurname, PersonPatronymic - имя, фамилия, отчество, заявителя.

ReceiptRegionName, ReceiptRegionCode - место получения справки, наименование и код.

StatementTypeName, StatementTypeCode - тип услуги, наименование и код.

StatementTypeSumOfService - сумма услуги.

ReceivingDate, ReceivingComment – дата/время, сообщение, когда будет готова справка (поля предназначены только для модуля «3 часа»), поля для экономических агентов, которые, в свою очередь, будут отображать их в удобной для клиента форме. Клиенту в свою очередь в интерфейсе агента отобразятся правила принятия условий в виде отображения «ДА»/«НЕТ». В случае, если клиент согласен, пропускается оплата.

b)

```
Code = 1;  
Name = "ServiceError";
```

Нет подключения или внутренняя ошибка!  
ObjectResult = null

c)

```
Code = 2;  
Name = "InvalidCompanyIdentifier ";
```

Неверный идентификатор!  
ObjectResult = null

d)

```
Code = 4;  
Name = "DataIdentifierIsEmpty ";
```

Идентификатор данных пустой!  
ObjectResult = null

e)

```
Code = 5;  
Name = "DataIdentifierMustContain16Characters";
```

Идентификатор данных должен содержать 16 цифр!  
ObjectResult = null

f)

```
Code = 6;  
Name = "StatementNotFound";
```

Не были найдены данные заявления!  
ObjectResult = null

e)

```
Code = 7;  
Name = "DocumentAlreadyPaid";
```

Документ был оплачен!  
ObjectResult = null

f)

Code = x;

Name = "Другой статус";

Другие возникшие статусы во время использования сервиса. В этом случае:

ObjectResult = null

**Пример запроса:**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xro="http://x-road.eu/xsd/xroad.xsd" xmlns:iden="http://x-road.eu/xsd/identifiers"
xmlns:erec="http://erecordV2_service.x-road.ee/erecordV2">
  <soapenv:Header>
    <xro:protocolVersion>4.0</xro:protocolVersion>
    <xro:issue></xro:issue>
    <xro:id></xro:id>
    <xro:userId></xro:userId>
    <xro:service iden:objectType="SERVICE">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>GOV</iden:memberClass>
      <iden:memberCode>70000006</iden:memberCode>
      <iden:subsystemCode>record-service</iden:subsystemCode>
    <iden:serviceCode>ERecordV2StatementVerifyDataByBarcodeTest</iden:serviceCode>
    <iden:serviceVersion>v1</iden:serviceVersion>
  </xro:service>
    <xro:client iden:objectType="SUBSYSTEM">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>YouMemberClass</iden:memberClass>
      <iden:memberCode>YouMemberCode</iden:memberCode>
      <iden:subsystemCode>YouSubsystemCode</iden:subsystemCode>
    </xro:client>
  </soapenv:Header>
  <soapenv:Body>
    <erec:ERecordV2StatementVerifyDataByBarcodeTest>
      <erec:companyIdentifier>0000000-0000-0000-0000-0000-000000000</erec:companyIdentifier>
      <erec:dataIdentifier>4020000000400000</erec:dataIdentifier>
    </erec:ERecordV2StatementVerifyDataByBarcodeTest>
  </soapenv:Body>
</soapenv:Envelope>

```

*ERecordV2StatementConfirmPaymentResponse* (*ERecordV2StatementConfirmPaymentRequest* *request*) - «Е-Заявление» - Подтверждение оплаты.

А также:

*ERecordV2StatementConfirmPaymentTestResponse*  
*ERecordV2StatementConfirmPaymentTest*  
 (*ERecordV2StatementConfirmPaymentTestRequest request*) - «Е-Заявление» (тестовая среда) - Подтверждение оплаты.

Где:

*request* – параметры запроса с заголовком СМЭВ Түндүк и дополнительные параметры;

*companyIdentifier* - полученный уникальный персональный идентификатор;

*secretKey* - идентификатор подтверждения оплаты из объекта результата метода *ERecordV2StatementVerifyDataByPin* (*ERecordV2StatementVerifyDataByPinTest*), *ERecordV2StatementVerifyDataByBarcode* (*ERecordV2StatementVerifyDataByBarcodeTest*);

*dataIdentifier* - идентификатор данных из объекта результата метода *ERecordV2StatementVerifyDataByPin* (*ERecordV2StatementVerifyDataByPinTest*), *ERecordV2StatementVerifyDataByBarcode* (*ERecordV2StatementVerifyDataByBarcodeTest*);

*paymentRegionCode* – область проведения оплаты, код из реестра -

Бишкек	1
Ош	2
Каракол	3
Нарын	4
Талас	5
Баткен	6
Жалал-Абад	7

Сервис возвращает:

а)

Code = 0;  
 Name = "Success";

Успешно, в этом случае возвращается объект *ObjectResult*, где:

*PaymentIdentifier* - идентификатор платежа, используется при отмене платежа;

DocumentCode - 16 значный код – идентификатор заявления/справки (**важно: нужно указать данный идентификатор в квитанции**), используется при апостилировании справок.

PersonName, PersonSurname, PersonPatronymic – имя, фамилия, отчество, заявителя.

ReceiptRegionName, ReceiptRegionCode - место получения справки, наименование и код.

PaymentRegionName, PaymentRegionCode - область проведения оплаты, наименование и код.

StatementTypeName, StatementTypeCode - тип услуги, наименование и код.

ReceivingDate - дата и время, когда будет готова справка.

ReceivingComment – сообщение, когда будет готова справка.

Receiver – Получатель (относится к банковским реквизитам);

TransitAccount - Транзитный счет (относится к банковским реквизитам);

AccumulationAccount - Лицевой счет (относится к банковским реквизитам);

BICBank - БИК банка (относится к банковским реквизитам);

ReceiverINN – ИНН получателя (относится к банковским реквизитам);

GNI – ГНИ (относится к банковским реквизитам);

OKPO – ОКПО (относится к банковским реквизитам);

Bank – Банк (относится к банковским реквизитам);

b)

Code = 1;

Name = " **ServiceError**";

Нет подключения или внутренняя ошибка!

ObjectResult = **null**

c)

Code = 2;

Name = " **InvalidCompanyIdentifier** ";

Неверный идентификатор!

ObjectResult = **null**

d)

Code = 4;

Name = " **SecretKeyIsEmpty**";

Идентификатор заявления пустой!

ObjectResult = **null**

e)

Code = 5;  
Name = "PaymentError";

Ошибка проведения платежа!  
ObjectResult = null

f)

Code = 6;  
Name = "DataIdentifierMustContain16Characters";

Идентификатор данных должен содержать 16 цифр!  
ObjectResult = null

g)

Code = 11;  
Name = "PaymentRegionNotFound ";

Указанное место проведения платежа не существует!  
ObjectResult = null

h)

Code = x;  
Name = "Другой статус";

Другие возникшие статусы во время использования сервиса. В этом случае:  
ObjectResult = null

### Пример запроса:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xro="http://x-road.eu/xsd/xroad.xsd" xmlns:iden="http://x-road.eu/xsd/identifiers"
xmlns:erec="http://erecordV2_service.x-road.ee/erecordV2">
  <soapenv:Header>
    <xro:protocolVersion>4.0</xro:protocolVersion>
    <xro:issue></xro:issue>
    <xro:id></xro:id>
    <xro:userId></xro:userId>
    <xro:service iden:objectType="SERVICE">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>GOV</iden:memberClass>
      <iden:memberCode>70000006</iden:memberCode>
```

```

    <iden:subsystemCode>record-service</iden:subsystemCode>
    <iden:serviceCode>ERecordV2StatementConfirmPaymentTest</iden:serviceCode>
    <iden:serviceVersion>v1</iden:serviceVersion>
  </xro:service>
  <xro:client iden:objectType="SUBSYSTEM">
    <iden:xRoadInstance>central-server</iden:xRoadInstance>
    <iden:memberClass>YouMemberClass</iden:memberClass>
    <iden:memberCode>YouMemberCode</iden:memberCode>
    <iden:subsystemCode>YouSubsystemCode</iden:subsystemCode>
  </xro:client>
</soapenv:Header>
<soapenv:Body>
  <erec:ERecordV2StatementConfirmPaymentTest>
    <erec:companyIdentifier>0000000-0000-0000-0000-
000000000</erec:companyIdentifier>
    <erec:secretKey>3c853fe4-a368-48f8-8240-73e2251efb4c</erec:secretKey>
    <erec:dataIdentifier>0000667420004292</erec:dataIdentifier>
    <erec:paymentRegionCode>2</erec:paymentRegionCode>
  </erec:ERecordV2StatementConfirmPaymentTest>
</soapenv:Body>
</soapenv:Envelope>

```

**ERecordV2CancelConfirmedPaymentResponse**      **ERecordV2CancelConfirmedPayment**  
**(ERecordV2CancelConfirmedPaymentRequest request)** - «Е-Заявление» - Отмена оплаты  
и заявления.

А также:

**ERecordV2CancelConfirmedPaymentTestResponse**  
**ERecordV2CancelConfirmedPaymentTest**  
**(ERecordV2CancelConfirmedPaymentTestRequest request)** - «Е-Заявление» (тестовая  
среда) - Отмена оплаты и заявления.

Где:

request – параметры запроса с заголовком СМЭВ Түндүк и дополнительные  
параметры;  
companyIdentifier - полученный уникальный персональный идентификатор;  
paymentIdentifier – идентификатор платежа, полученный при проведении платежа  
из объекта результата метода **ERecordV2StatementConfirmPayment**  
**(ERecordV2StatementConfirmPaymentTest)**;

documentCode - 16 значный код документа полученный при проведении платежа из объекта результата метода *ERecordV2StatementConfirmPayment* (*ERecordV2StatementConfirmPaymentTest*);

Сервис возвращает:

a)

Code = 0;  
Name = "Success";

ObjectResult возвращает true, платеж успешно отменен.

b)

Code = 1;  
Name = " ServiceError";  
Нет подключения или внутренняя ошибка!  
ObjectResult = false

c)

Code = 2;  
Name = "InvalidCompanyIdentifier";  
Неверный идентификатор!  
ObjectResult = false

d)

Code = 4;  
Name = " DocumentCodeIsEmpty";

Не был указан 16 значный код документа!  
ObjectResult = false

e)

Code = 5;  
Name = "DocumentCodeMustContain16Characters ";

Код документа должен содержать 16 цифр!  
ObjectResult = false

f)

Code = 6;  
Name = " StatementNotFoundToCancelPayment";

Не было найдено заявление!  
ObjectResult = false

g)

Code = 7;

Name = " **PaymentNotFound** ";

Не был найден платеж!

ObjectResult = **false**

h) Code = 8;

Name = " **PaymentIdentifierIsEmpty** ";

Идентификатор платежа пустой!

ObjectResult = **false**

i) Code = 9;

Name = " **PaymentCanNotBeCancellation** ";

Заявление взято в обработку, не подлежит отмене!

ObjectResult = **false**

h) Code = x;

Name = " **Другой статус** ";

Другие возникшие статусы во время использования сервиса. В этом случае:

ObjectResult = **null****Пример запроса:**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xro="http://x-road.eu/xsd/xroad.xsd" xmlns:iden="http://x-road.eu/xsd/identifiers"
xmlns:erec="http://erecordV2_service.x-road.ee/erecordV2">
  <soapenv:Header>
    <xro:protocolVersion>4.0</xro:protocolVersion>
    <xro:issue></xro:issue>
    <xro:id></xro:id>
    <xro:userId></xro:userId>
    <xro:service iden:objectType="SERVICE">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>GOV</iden:memberClass>
      <iden:memberCode>70000006</iden:memberCode>
      <iden:subsystemCode>record-service</iden:subsystemCode>
      <iden:serviceCode>ERecordV2CancelConfirmedPaymentTest</iden:serviceCode>
      <iden:serviceVersion>v1</iden:serviceVersion>
    </xro:service>
    <xro:client iden:objectType="SUBSYSTEM">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>

```

```

    <iden:memberClass>YouMemberClass</iden:memberClass>
    <iden:memberCode>YouMemberCode</iden:memberCode>
    <iden:subsystemCode>YouSubsystemCode</iden:subsystemCode>
  </xro:client>
</soapenv:Header>
<soapenv:Body>
  <erec:ERecordV2CancelConfirmedPaymentTest>
    <erec:companyIdentifier>00000000-0000-0000-0000-
0000000000</erec:companyIdentifier>
    <erec:paymentIdentifier>37590f66-fd55-4b8f-8036-
7b4c57ca41da</erec:paymentIdentifier>
    <erec:documentCode>2006329900060000</erec:documentCode>
  </erec:ERecordV2CancelConfirmedPaymentTest>
</soapenv:Body>
</soapenv:Envelope>

```

*ERecordV2ListPlacesOfReceiveResponse*

*ERecordV2ListPlacesOfReceive*

(*ERecordV2ListPlacesOfReceiveRequest request*) - «Е-Заявление» - место получения справок согласно типа услуг.

Где:

request – параметры запроса с заголовком СМЭВ Түндүк и дополнительные параметры;

statementTypeCode - тип услуги, код из реестра -

1 день	1
3 дня	2
7 дней	3
3 часа	4

Метод сервиса возвращает:

Сервис возвращает:

Status – статус обработки данных:

a) Code = 0;

Name = "Success";

Успешно. В этом случае возвращается массив из объектов ObjectResult, где:

Name – наименование.

Code – код.

b) Code = 1;

Name = " ServiceError";

Нет подключения или внутренняя ошибка!

ObjectResult = null

c)Code = 2;  
 Name = " **StatementTypeNotFound** ";  
 Указанный тип услуги не существует!  
 ObjectResult = **null**

d)Code = 4;  
 Name = " **EmptyList** ";  
 Список пуст!  
 ObjectResult = **null**

### Пример запроса:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xro="http://x-road.eu/xsd/xroad.xsd" xmlns:iden="http://x-road.eu/xsd/identifiers"
xmlns:erec="http://erecordV2_service.x-road.ee/erecordV2">
  <soapenv:Header>
    <xro:protocolVersion>4.0</xro:protocolVersion>
    <xro:issue></xro:issue>
    <xro:id></xro:id>
    <xro:userId></xro:userId>
    <xro:service iden:objectType="SERVICE">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>GOV</iden:memberClass>
      <iden:memberCode>70000006</iden:memberCode>
      <iden:subsystemCode>record-service</iden:subsystemCode>
      <iden:serviceCode>ERecordV2ListPlacesOfReceive</iden:serviceCode>
      <iden:serviceVersion>v1</iden:serviceVersion>
    </xro:service>
    <xro:client iden:objectType="SUBSYSTEM">
      <iden:xRoadInstance>central-server</iden:xRoadInstance>
      <iden:memberClass>YouMemberClass</iden:memberClass>
      <iden:memberCode>YouMemberCode</iden:memberCode>
      <iden:subsystemCode>YouSubsystemCode</iden:subsystemCode>
    </xro:client>
  </soapenv:Header>
  <soapenv:Body>
    <erec:ERecordV2ListPlacesOfReceive>
      <erec:statementTypeCode>1</erec:statementTypeCode>
    </erec:ERecordV2ListPlacesOfReceive>
  </soapenv:Body>
</soapenv:Envelope>
```