Сделано ИИ

- Поручение экспедитору
- Единый транспортный документ
- Взятие в ответственность ЭПЭ
- ФСБ

Поручение экспедитору

Схемы форматов Поручения экспедитору и расписок доступны на сайте ФНС по ссылке

Почему в комплекте 3 xsd схемы ON_POREXPEXP_1_958_06_05_01_01.xsd ON_POREXPKLT_1_958_05_05_01_01.xsd ON_POREXPOTZ_1_958_07_05_01_01.xsd какая за что отвечает?

в комплекте действительно **три XSD-схемы**, и это не случайно. Они соответствуют **трём разным файлам обмена**, которые вместе составляют жизненный цикл электронного поручения экспедитору в ГИС ЭПД.

Вот подробная расшифровка, какая схема за что отвечает, с прямыми ссылками на ваш файл.

---

1. ON_POREXPKLT_1_958_05_05_01_01.xsd — **Информация клиента** (первичное поручение)

Параметр	Значение
Префикс	`ON_POREXPKLT` (расшифровывается как **Por**uchenie **Exp**editorskoe **K**lient — Поручение Экспедитору, Клиент)
За что отвечает	Это **основной документ**, который создаёт **клиент** (заказчик). В нём содержится вся информация о перевозке: данные о грузе, маршруте, сторонах (грузоотправитель, грузополучатель, экспедитор), условиях, страховке и т.д.
Кто подписывает	Клиент (отправитель файла)
Ссылка в вашем файле	Раздел **II** (стр. 2–33), таблицы 5.1–5.49. Имя схемы указано в пункте 4: `ON_POREXPKLT_1_958_05_05_01_xx`

Это **"оферта"** от клиента экспедитору.

---

2. ON_POREXPEXP_1_958_06_05_01_01.xsd — **Информация экспедитора** (согласование)

Параметр	Значение
Префикс	`ON_POREXPEXP` (**Por**uchenie **Exp**editorskoe **Exp**editor — Поручение Экспедитору, Экспедитор)
За что отвечает	Это **ответный документ**, который создаёт **экспедитор**. Он подтверждает (согласовывает) принятие поручения, указывает свой реестровый номер, дополнительные условия, информацию о привлечении третьих лиц, оплате и транспортировке.
Кто подписывает	Экспедитор (отправитель файла)
Ссылка в вашем файле	Раздел **III** (стр. 33–57), таблицы 7.1–7.37. Имя схемы указано в пункте 6: `ON_POREXPEXP_1_958_06_05_01_xx`

Это **"акцепт"** от экспедитора клиенту. Без этого файла электронное поручение считается недействительным (п. 2 вашего документа).

---

3. ON_POREXPOTZ_1_958_07_05_01_01.xsd — **Отзыв поручения клиентом**

Параметр	Значение
Префикс	`ON_POREXPOTZ` (**Por**uchenie **Exp**editorskoe **Otz**yv — Поручение Экспедитору, Отзыв)
За что отвечает	Это **опциональный документ**, который создаёт **клиент**, если он хочет отозвать уже выданное поручение экспедитору. В нём указывается причина отзыва, дата прекращения исполнения, обязательство возместить фактические расходы экспедитору и инструкции по возврату груза.
Кто подписывает	Клиент
Ссылка в вашем файле	Раздел **IV** (стр. 57–69), таблицы 9.1–9.23. Имя схемы указано в пункте 8: `ON_POREXPOTZ_1_958_07_05_01_xx`

Это **"аннулирование"** или отказ от поручения.

Схема взаимодействия (последовательность)

text

```
1. Клиент формирует XML по схеме ON_POREXPKLT → подписывает → отправляет экспедитору
   ↓
2. Экспедитор получает, проверяет, формирует XML по схеме ON_POREXPEXP → подписывает → отправляе
   ↓
3. (Если нужно) Клиент формирует XML по схеме ON_POREXPOTZ → подписывает → отправляет экспедитор
```

□□ Почему три, а не одна?

Потому что электронное поручение — это **двухсторонний (а в случае отзыва — трёхсторонний в смысле стадий) документооборот**. Каждая сторона создаёт **свой отдельный XML-файл** со своей подписью. Это позволяет:

- Чётко разделить ответственность (кто что внёс);
- Обеспечить юридическую значимость каждой стадии;
- ГИС ЭПД может сопоставить все три файла по общему **УИД (уникальному идентификатору)** и собрать целостную картину сделки.

Где искать XSD-файлы на сайте ФНС?

В приказе прямо указано (п. 4, 6, 8):

> "*XML-схема файла обмена в электронной форме приводится отдельным файлом и размещается на официальном сайте Федеральной налоговой службы в информационно-телекоммуникационной сети «Интернет».*"

Обычно они публикуются в разделе **"Электронные услуги" → "Электронный документооборот" → "Утверждённые форматы"** или в карточке самого приказа в виде ZIP-архива с приложениями. Если на странице приказа их нет — обратитесь к своему оператору ЭДО, они уже должны иметь актуальные версии для настройки интеграции.

Покажи каждый из случаев на примере, согласно указанным схемам

Отлично, теперь, имея на руках сами XSD-схемы, мы можем показать **реальные XML-фрагменты** для каждого из трёх файлов обмена. Ниже приведены **минимальные валидные примеры** (согласно логике схем), которые показывают структуру и ключевые элементы.

Случай 1. Клиент создаёт поручение (ON_POREXPKLT)

Кто: Клиент (грузоотправитель или заказчик).
Цель: Отправить экспедитору заявку на организацию перевозки.

xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Файл ИдФайл="ON_POREXPKLT_1234567890_20260601_550e8400-e29b-41d4-a716-446655440000"
      ВерсПрог="1.0"
      ВерсФорм="5.01">
  <Документ КНД="1110486"
            НаимДок="электронное поручение экспедитору, информация клиента"
            ДатИнфКлнт="01.06.2026"
            ВрИнфКлнт="12:00:00">
    <СодИнфКлнт СодОпер="представление электронного поручения экспедитору"
                НомДок="ПЭ-001"
                ДатаДок="01.06.2026"
                ДокГруз="0">
      <!-- Сведения о грузе -->
      <СвГруз ПрУведСторон="0"
              ДатОтгрС="01.06.2026"
              ДатОтгрПо="02.06.2026"
              ПрзПрвз="0"
              ОтмПрием="0">
        <СвГП>
          <ИдСв>
            <СвЮЛУч НаимОрг="ООО Получатель"
                    ИННЮЛ="1234567890"/>
          </ИдСв>
        </СвГП>
        <СвГО>
          <ИдСв>
            <СвЮЛУч НаимОрг="ООО Отправитель"
                    ИННЮЛ="0987654321"/>
          </ИдСв>
        </СвГО>
        <СвТС ВидТС="1"/> <!-- автотранспорт -->
        <ОпГруз НаимГруз="Запчасти"
                КолМестГр="10"
                ПрзОпаснГруз="0"
                ПрзИзъятВещ="0"
                УказОбъем="0"
                УчГосСист="0"
                НалУпак="0"
                НалКодТовНом="0">
          <Марк>Отсутствует</Марк>
```

```
      <СвСтрПроисх КодСтр="643"
                   НаимСтран="Россия"/>
      <ВесГруз ВесНетто="250.000"
               ВесБрутто="260.000"/>
    </ОпГруз>
    <АдрГруз>
      <АдрПриемГруз>
        <Адрес>
          <АдрРФ КодРегион="77"
                 НаимРегион="г. Москва"/>
        </Адрес>
      </АдрПриемГруз>
    </АдрГруз>
    <АдрПунктНазн>
      <АдрВыдГруз>
        <Адрес>
          <АдрРФ КодРегион="78"
                 НаимРегион="г. Санкт-Петербург"/>
        </Адрес>
      </АдрВыдГруз>
    </АдрПунктНазн>
  </СвГруз>
  <!-- Участники -->
  <СвКлнт>
    <ИдСв>
      <СвЮЛУч НаимОрг="ООО Клиент"
              ИННЮЛ="1111111111"/>
    </ИдСв>
  </СвКлнт>
  <СвЭксп>
    <ИдСв>
      <СвЮЛУч НаимОрг="ООО Экспедитор"
              ИННЮЛ="2222222222"/>
    </ИдСв>
  </СвЭксп>
  <!-- Договор -->
  <ДогТрЭксп РеквНаимДок="Договор транспортной экспедиции"
             РеквНомерДок="Д-001"
             РеквДатаДок="01.01.2026"/>
</СодИнфКлнт>
<!-- Подпись клиента -->
```

```
    <ПодпИнфКлнт СпосПодтПолном="1">
      <ФИО Фамилия="Иванов"
           Имя="Иван"
           Отчество="Иванович"/>
    </ПодпИнфКлнт>
  </Документ>
</Файл>
```

Случай 2. Экспедитор согласует (ON_POREXPEXP)

Кто: Экспедитор.
Цель: Подтвердить принятие поручения, указать свои условия и реестровый номер.

xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Файл ИдФайл="ON_POREXPEXP_0987654321_20260601_550e8400-e29b-41d4-a716-446655440001"
      ВерсПрог="1.0"
      ВерсФорм="5.01">
  <Документ КНД="1110487"
            НаимДок="электронное поручение экспедитору, информация экспедитора"
            ДатИнфЭксп="01.06.2026"
            ВрИнфЭксп="14:30:00">
    <!-- Ссылка на файл клиента -->
    <ИдИнфКлнт ИдФайлИнфКлнт="ON_POREXPKLT_1234567890_20260601_550e8400-e29b-41d4-a716-
446655440000"
               ДатФайлИнфКлнт="01.06.2026"
               ВрФайлИнфКлнт="12:00:00">
      <ЭП>base64-подпись-клиента</ЭП>
    </ИдИнфКлнт>
    <!-- Согласование экспедитора -->
    <СодИнфЭксп УИД_ПорЭксп="550e8400-e29b-41d4-a716-446655440000"
                НомРеестр="Р-001"
                ДатаРеестр="01.06.2026"
                СодОпер="согласование подлежащих оказанию транспортно-экспедиционных услуг и
условий">
      <!-- Дополнительная информация об услугах -->
      <ДопИнфЭксп ПризнИсп="1"
```

```
                  НаимУсл="Организация автоперевозки">
        <СвОплата ОбщСтУсл="50000.00"
                  НалСт="без НДС"
                  КодОКВ="643"
                  НаимОКВ="Российский рубль"/>
        <СвТрсГруз МаршСлГруз="Москва → Санкт-Петербург">
          <СвТСДост ВидТС="1">
            <СвВодит НомВУ="1234567890"
                     СерВУ="78"
                     ДатаВыдВУ="01.01.2020">
              <Тлф>+79111234567</Тлф>
              <ФИО Фамилия="Петров"
                   Имя="Петр"
                   Отчество="Петрович"/>
            </СвВодит>
          </СвТСДост>
        </СвТрсГруз>
      </ДопИнфЭксп>
    </СодИнфЭксп>
    <!-- Подпись экспедитора -->
    <ПодпИнфЭксп СпосПодтПолном="1">
      <ФИО Фамилия="Сидоров"
           Имя="Сергей"
           Отчество="Александрович"/>
    </ПодпИнфЭксп>
  </Документ>
</Файл>
```

Случай 3. Клиент отзывает поручение (ON_POREXPOTZ)

Кто: Клиент.
Цель: Отозвать ранее выданное поручение и уведомить экспедитора.

xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Файл ИдФайл="ON_POREXPOTZ_1122334455_20260602_550e8400-e29b-41d4-a716-446655440002"
      ВерсПрог="1.0"
```

```
      ВерсФорм="5.01">
  <Документ КНД="1110488"
            НаимДок="электронное поручение экспедитору, информация клиента об отзыве поручения
экспедитору"
            ДатИнфОтз="02.06.2026"
            ВрИнфОтз="09:00:00">
    <!-- Ссылка на файл клиента (первичное поручение) -->
    <ИдИнфКлнт ИдФайлИнфКлнт="ON_POREXPKLT_1234567890_20260601_550e8400-e29b-41d4-a716-
446655440000"
               ДатФайлИнфКлнт="01.06.2026"
               ВрФайлИнфКлнт="12:00:00">
      <ЭП>base64-подпись-клиента-первичная</ЭП>
    </ИдИнфКлнт>
    <!-- Содержание отзыва -->
    <СодИнфОтз УИД_ПорЭксп="550e8400-e29b-41d4-a716-446655440000"
               СодОпер="отзыв поручения экспедитору и обязательство возместить экспедитору
фактические расходы"
               ДатаОтз="02.06.2026"
               СумФактРасх="1500.00">
      <СвВозврГруз УслВозврГруз="2"> <!-- забирает клиент -->
        <КонтЛицо Фамилия="Иванов"
                  Имя="Иван"
                  Отчество="Иванович"/>
        <Контакт>
          <Тлф>+79119876543</Тлф>
        </Контакт>
        <АдрПунктВозвр>
          <Адрес>
            <АдрРФ КодРегион="77"
                   НаимРегион="г. Москва"/>
          </Адрес>
        </АдрПунктВозвр>
      </СвВозврГруз>
    </СодИнфОтз>
    <!-- Подпись клиента -->
    <ПодпИнфОтз СпосПодтПолном="1">
      <ФИО Фамилия="Иванов"
           Имя="Иван"
           Отчество="Иванович"/>
    </ПодпИнфОтз>
  </Документ>
```

```
</Файл>
```

□□ Ключевые отличия в структуре (по схемам)

Что проверяем	ON_POREXPKLT (клиент)	ON_POREXPEXP (экспедитор)	ON_POREXPOTZ (отзыв)
КНД	`1110486`	`1110487`	`1110488`
Содержание операции	`представление электронного поручения экспедитору`	`согласование подлежащих оказанию ...`	`отзыв поручения экспедитору ...`
Обязательные реквизиты	Номер поручения, дата, груз, стороны, договор	УИД поручения, номер и дата реестра	УИД поручения, дата отзыва
Вложенные блоки	`СвГруз`, `СвКлнт`, `СвЭксп`, `ДогТрЭксп`	`ИдИнфКлнт` (ссылка на файл клиента + его ЭП), `ДопИнфЭксп`	`ИдИнфКлнт`, `СвВозврГруз`
Признак для экспедитора	Нет	`ПризнИсп` (1/2/3)	Нет

□□ Что важно понимать про эти примеры

1. **Все примеры минимальны** — в реальности поля будут заполнены полнее (адреса, банковские реквизиты, контактные данные и т.д.).
2. **Связка по УИД** — во втором и третьем файлах обязательно указывается `УИД_ПорЭксп`, полученный от ГИС ЭПД или из первого файла.
3. **Подписи** — в примерах использован `СпосПодтПолном="1"` (подпись на основании данных в самой ЭП). Если нужна доверенность — нужно использовать `СпосПодтПолном="3"` или `"5"` и вложить блоки `СвДоверЭл` или `СвДоверБум`.

Единый транспортный документ

XSD - схемы на сайте ФТС

В Приказ ФНС России от 13.08.2025 № ЕД-7-26/717@ Два файла со схемами: xsd-схемы xsd-схемы действующие с 01.01.2026 (версия от 31.10.2025) Почему их две. Какие актуальные?

Наличие двух версий XSD-схем для Приказа ФНС № ЕД-7-26/717@ — это стандартная практика ФНС. Вторая версия (от 31.10.2025) является **актуальной**, и применять её нужно с **1 января 2026 года**.

Почему существуют две версии схем?

Это часть обычного рабочего процесса ФНС по внедрению нового формата электронного документа.

1. **Первая версия (исходная)**: Утверждена приказом от 13.08.2025 и зарегистрирована в Минюсте 15.09.2025. Она описывает базовые форматы для нового электронного единого транспортного документа (ЕТД).

2. **Вторая версия (актуализированная)**: Появляется позже — версия от 31.10.2025. **Главная причина** — внесение изменений, необходимых из-за поправок в Налоговый кодекс. С 1 января 2026 года ставка НДС повышается до 22%. ФНС оперативно обновляет все свои XSD-схемы, чтобы в XML-файлах можно было указывать новую ставку налога. Схемы для ЕТД, как и для счетов-фактур, были скорректированы для этой цели.

Какая версия актуальна?

Актуальной является версия от 31.10.2025.

- **С 1 января 2026 года** для создания электронных ЕТД нужно использовать **только обновленную схему (версии от 31.10.2025)**, поскольку она учитывает новую ставку НДС.
- До этой даты (в 2025 году) можно было работать и с исходной схемой, утвержденной в августе.

Таким образом, для отчетности в 2026 году следует применять XSD-схему версии от 31.10.2025.

Если вам нужна помощь в поиске конкретных файлов или в разборе структуры обновленной схемы — напишите.

Загружаю все xsd схемы. Прошу составить минимальный пример мультимодальной перевозки. По морю, Шанхай - Владивосток; по ЖД - Владивосток - Москва; на авто - до склада получателя

Отлично, вы предоставили полный набор XSD-схем для электронного единого транспортного документа (ЭЕТД). На их основе я составлю минимальный, но полный пример мультимодальной перевозки по вашему маршруту.

Важно: Согласно Приказу ФНС, для мультимодальной перевозки информация передается частями разными участниками. Ваш полный маршрут (Шанхай -> Владивосток -> Москва -> Склад) будет описан в **трех** основных файлах обмена, создаваемых разными сторонами:

1. **Информация Грузоотправителя (`ON_ETDGO_1_959_01_05_01_01.xsd`)** — создается отправителем в Шанхае. Содержит общие данные о грузе, грузополучателе, договоре и т.д. Это "шапка" документа.
2. **Информация Перевозчика (`ON_ETDPRV_1_959_02_05_01_01.xsd`)** — создается перевозчиком (например, логистической компанией). Содержит **полный маршрут (`МрштПрвз`)** из нескольких участков и детали по каждому виду транспорта.
3. **Информация Грузополучателя (`ON_ETDGRPO_1_959_07_05_01_01.xsd`)** — создается конечным получателем в Москве. Подтверждает факт приема груза.

Ниже приведены минимальные XML-фрагменты для этих файлов.

---

1. Информация Грузоотправителя (Файл: `ON_ETDGO...`)

Этот файл создается отправителем в Шанхае. В нем указываются основные реквизиты сторон и груза.

xml

```
<?xml version="1.0" encoding="windows-1251"?>
<Файл ИдФайл="ETD_GO_20250624_001" ВерсПрог="1.0" ВерсФорм="5.01"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <ИдПолИной>ID_OTHER_RECIPIENT_001</ИдПолИной>
    <Документ КНД="1110452" НаимДокИнфГО="электронный единый транспортный документ, информация грузоотправителя"
              ДатИнфГО="24.06.2026" ВрИнфГО="10:00:00">
        <СодИнфГО УИД_ЕТД="12345678-1234-1234-1234-123456789012" НомДок="ETD-2026-001" ДатаДок="24.06.2026"
                  НомДогПер="CTR-2026-001" ДатаДогПер="01.06.2026">

            <!-- Сведения о грузоотправителе (Shanghai Exporter) -->
            <СвГО ГОЭксп="0">
                <РекИдентГО>
                    <ИдСв>
                        <СвЮЛУч ИННЮЛ="1234567890" ОГРН="1234567890123" НаимОрг="Shanghai Export Co., Ltd"/>
                    </ИдСв>
```

```
            </РекИдентГО>
        </СвГО>

        <!-- Сведения о грузополучателе (Moscow Importer) -->
        <СвГП ГПСт="1">
            <РекИдентГП>
                <ИдСв>
                    <СвЮЛУч ИННЮЛ="0987654321" ОГРН="3210987654321" НаимОрг="Moscow Import
LLC"/>
                </ИдСв>
            </РекИдентГП>
            <!-- Документ на получение груза -->
            <ОснЛицГП РеквНаимДок="Доверенность" РеквНомерДок="PWR-2026-01"
РеквДатаДок="01.06.2026"/>
        </СвГП>

        <!-- Сведения о перевозчике (Carrier) -->
        <СвПрв>
            <ИдСв>
                <СвЮЛУч ИННЮЛ="1122334455" ОГРН="5544332211001" НаимОрг="Global Logistics
Carrier"/>
            </ИдСв>
        </СвПрв>

        <!-- Сведения о грузе -->
        <СвГруз>
            <ОпГруз НомерГруз="1" НаимГруз="Электроника" КолМестГр="10" УчГосСист="0">
                <Марк>Маркировка 001</Марк>
                <Марк>Маркировка 002</Марк>
                <!-- Хотя бы один из ВесГруз или ВесГрузНетто должен быть -->
                <СвГосСист НаимГосСист="Маркировка" ИдНомУчетЕд="ID-12345" УчетЕд="шт"/>
                <ВесГруз>1000.00</ВесГруз>
            </ОпГруз>
            <ОбЦеннГр>
                <СтЦеннГр>50000.00</СтЦеннГр>
                <КодОКВ>840</КодОКВ>
                <НаимОКВ>USD</НаимОКВ>
            </ОбЦеннГр>
        </СвГруз>

        <!-- Пункты отправления и назначения (общие, из договора) -->
```

```
            <СвПункт НаимПунктОтпр="Shanghai Port" НаимПунктНазн="Moscow Warehouse">
                <АдрПунктОтпр>
                    <Адрес>
                        <АдрИнф КодСтр="156" НаимСтр="China" АдрТекст="Shanghai Port"/>
                    </Адрес>
                </АдрПунктОтпр>
                <АдрПунктНазн>
                    <Адрес>
                        <АдрРФ КодРегион="77" НаимРегион="г. Москва" Город="Москва" Улица="ул.
Логистическая" Дом="10"/>
                    </Адрес>
                </АдрПунктНазн>
            </СвПункт>
        </СодИнфГО>

        <!-- Подпись отправителя -->
        <ПодпИнфГО Должн="Директор" СпосПодтПолном="1">
            <ФИО Фамилия="Иванов" Имя="Иван" Отчество="Иванович"/>
        </ПодпИнфГО>
    </Документ>
</Файл>
```

---

2. Информация Перевозчика (Файл: `ON_ETDPRV...`)

Это ключевой файл, где описывается **полный маршрут из трех участков**, согласованный с информацией от грузоотправителя. Здесь мы детализируем каждый вид транспорта.

xml

```
<?xml version="1.0" encoding="windows-1251"?>
<Файл ИдФайл="ETD_PRV_20250624_001" ВерсПрог="1.0" ВерсФорм="5.01"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <ИдПолИной>ID_OTHER_RECIPIENT_001</ИдПолИной>
    <Документ КНД="1110453" НаимДокИнфПрв="электронный единый транспортный документ,
информация перевозчика"
              ДатИнфПрв="24.06.2026" ВрИнфПрв="12:00:00">

        <!-- Ссылка на файл грузоотправителя -->
```

```
        <ИдИнфГО ИдФайлИнфГО="ETD_GO_20250624_001" ДатФайлИнфГО="24.06.2026"
ВрФайлИнфГО="10:00:00" ЭП="SignatureGO"/>

        <СодИнфПрв УИД_ЕТД="12345678-1234-1234-1234-123456789012">

            <!-- 1. Полный маршрут перевозки (из 3-х участков) -->
            <МршtПрвз ОбщКолУч="3">
                <МршtСлед МршТекст="Шанхай (море) - Владивосток (ЖД) - Москва (авто)"/>
            </МршtПрвз>

            <!-- 2. Описание условий перевозки по каждому участку -->

            <!-- Участок 1: Море (Шанхай -> Владивосток) -->
            <УслПрвз НомУч="1" ВидТрс="5" СрокДостГр="7"
ДатВрПогр="25.06.2026T00:00:00+00:00">
                <ДатВыгр>02.07.2026</ДатВыгр>
                <АдрПунктОтпр>
                    <Адрес>
                        <АдрИнф КодСтр="156" НаимСтр="China" АдрТекст="Shanghai Port"/>
                    </Адрес>
                </АдрПунктОтпр>
                <АдрПунктНазн>
                    <Адрес>
                        <АдрРФ КодРегион="25" НаимРегион="Приморский край" Город="Владивосток"
Улица="ул. Портовая" Дом="1"/>
                    </Адрес>
                </АдрПунктНазн>
                <!-- Детали морской перевозки -->
                <ПрвзМорскТрс>
                    <СвПортПогрз НазвПорт="Shanghai Port"/>
                    <СвПортВыгрз НазвПорт="Vladivostok Port"/>
                    <СвСудно НазвСудно="Ever Given" КодСтрРегСудно="156" ТипСудно="Container
Ship"/>
                </ПрвзМорскТрс>
            </УслПрвз>

            <!-- Участок 2: Железная дорога (Владивосток -> Москва) -->
            <УслПрвз НомУч="2" ВидТрс="4" СрокДостГр="10"
ДатВрПогр="03.07.2026T00:00:00+00:00">
                <ДатВыгр>13.07.2026</ДатВыгр>
                <АдрПунктОтпр>
```

```
                    <Адрес>
                        <АдрРФ КодРегион="25" НаимРегион="Приморский край" Город="Владивосток"
Улица="ул. Железнодорожная" Дом="1"/>
                    </Адрес>
                </АдрПунктОтпр>
                <АдрПунктНазн>
                    <Адрес>
                        <АдрРФ КодРегион="77" НаимРегион="г. Москва" Город="Москва" Улица="ул.
Вокзальная" Дом="1"/>
                    </Адрес>
                </АдрПунктНазн>
                <!-- Детали ЖД перевозки -->
                <ПрвзЖдТрс>
                    <СвСтанцПогрз КодАдмЖД="11" НаимЖД="Дальневосточная" КодСтЖД="123"
НаимСтЖД="Владивосток-Грузовой"/>
                    <СвСтанцВыгрз КодАдмЖД="17" НаимЖД="Московская" КодСтЖД="456"
НаимСтЖД="Москва-Сортировочная"/>
                    <СвВагон РодВагон="Крытый" НомерВагон="12345678" КодСтрРегВагон="RU"/>
                </ПрвзЖдТрс>
            </УслПрвз>

            <!-- Участок 3: Автомобиль (Москва -> Склад получателя) -->
            <УслПрвз НомУч="3" ВидТрс="1" СрокДостГр="1"
ДатВрПогр="14.07.2026T00:00:00+00:00">
                <ДатВыгр>15.07.2026</ДатВыгр>
                <АдрПунктОтпр>
                    <Адрес>
                        <АдрРФ КодРегион="77" НаимРегион="г. Москва" Город="Москва" Улица="ул.
Вокзальная" Дом="1"/>
                    </Адрес>
                </АдрПунктОтпр>
                <АдрПунктНазн>
                    <Адрес>
                        <АдрРФ КодРегион="77" НаимРегион="г. Москва" Город="Москва" Улица="ул.
Логистическая" Дом="10"/>
                    </Адрес>
                </АдрПунктНазн>
                <!-- Детали авто перевозки -->
                <ПрвзАвтТрс>
                    <ТС РегНомер="А123ВВ" ТипВлад="1">
                        <ПарамТС Тип="Фура" Марка="Volvo" Модель="FH" Грузопод="20.00"
```

```
Вместим="80.00"/>
                    </ТС>
                    <СвВодит>
                        <Тлф>+7(999)123-45-67</Тлф>
                        <ФИО Фамилия="Петров" Имя="Петр" Отчество="Петрович"/>
                        <ИННФЛ>123456789012</ИННФЛ>
                    </СвВодит>
                    <СпецУслДвиж ИдСР="SR-2026-001"/>
                </ПрвзАвтТрс>
            </УслПрвз>

        </СодИнфПрв>

        <!-- Подпись перевозчика -->
        <ПодпИнфПрв Должн="Генеральный директор" СпосПодтПолном="1">
            <ФИО Фамилия="Сидоров" Имя="Сергей" Отчество="Сергеевич"/>
        </ПодпИнфПрв>
    </Документ>
</Файл>
```

---

3. Информация Грузополучателя (Файл: `ON_ETDGRPO...`)

Финальный файл, который создает получатель в Москве, подтверждая факт получения груза. В нем обязательно указывается ссылка на файл перевозчика.

xml

```
<?xml version="1.0" encoding="windows-1251"?>
<Файл ИдФайл="ETD_GRPO_20260624_001" ВерсПрог="1.0" ВерсФорм="5.01"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <ИдПолИной>ID_OTHER_RECIPIENT_001</ИдПолИной>
    <Документ КНД="1110458" НаимДокИнфГП="электронный единый транспортный документ, информация
грузополучателя"
              ДатИнфГП="15.07.2026" ВрИнфГП="16:00:00">

        <!-- Ссылка на файл перевозчика -->
        <ИдИнфПрв ИдФайлИнфПрв="ETD_PRV_20250624_001" ДатФайлИнфПрв="24.06.2026"
ВрФайлИнфПрв="12:00:00" ЭП="SignaturePRV"/>
```

```
        <СодИнфГП УИД_ЕТД="12345678-1234-1234-1234-123456789012" СодОпер="1">

            <!-- Сведения о приеме груза -->
            <ПриемГрузГП ДатПриб="15.07.2026" КолМестПриемЧ="10"
                         ОбщСвСост="Груз в целости" МасБрутЗначПрием="1000.00">
                <АдрВыгруз>
                    <Адрес>
                        <АдрРФ КодРегион="77" НаимРегион="г. Москва" Город="Москва" Улица="ул.
Логистическая" Дом="10"/>
                    </Адрес>
                </АдрВыгруз>

                <СвПринПоНаим НаимГруз="Электроника" СостГруз="Целая" МасБрутПрием="1000.00">
                    <ПерМарк>Маркировка 001</ПерМарк>
                    <ПерМарк>Маркировка 002</ПерМарк>
                </СвПринПоНаим>
            </ПриемГрузГП>

        </СодИнфГП>

        <!-- Подпись получателя -->
        <ПодпИнфГП Должн="Начальник склада" СпосПодтПолном="1">
            <ФИО Фамилия="Смирнова" Имя="Елена" Отчество="Викторовна"/>
        </ПодпИнфГП>
    </Документ>
</Файл>
```

Пояснения:

1. **Коды КНД:** Для каждого типа документа свой уникальный код (`1110452`, `1110453`, `1110458`).
2. **УИД_ЕТД:** Уникальный идентификатор всей перевозки. Он должен быть одинаковым во всех файлах.
3. **Ссылочная целостность:** Файл перевозчика ссылается на файл отправителя (`ИдИнфГО`), а файл получателя — на файл перевозчика (`ИдИнфПрв`). Это позволяет собрать полную картину перевозки.
4. **Участки перевозки:** В файле перевозчика участки нумеруются (`НомУч`) и для каждого указывается вид транспорта (`ВидТрс`). Для моря — `5`, для ЖД — `4`, для авто — `1`. В зависимости от вида транспорта заполняются соответствующие блоки (`ПрвзМорскТрс`, `ПрвзЖдТрс`, `ПрвзАвтТрс`).

5. **Даты и время:** Формат дат — `ДД.ММ.ГГГГ`, дат с временем — `ДД.ММ.ГГГГТЧЧ:ММ:СС+-ЧЧ:ММ`. Для упрощения я указал `+00:00` (UTC).
6. **Обязательные поля:** В реальном примере нужно заполнить все обязательные поля, включая такие как `ИдПолИной` (если есть), и детали адресов. В моем примере адреса и реквизиты приведены минимально, но достаточно для валидации.

Эти три файла представляют собой полный электронный документооборот для вашей мультимодальной перевозки.

Взятие в ответственность ЭПЭ

В каких случаях ЭПЭ составляется с взятием груза в ответственность и без взятия? Как это применяется в субэкспедировании

Электронное поручение экспедитору (ЭПЭ) оформляется абсолютно при любом сценарии привлечения экспедитора. Однако юридическое разделение на работу «с взятием под ответственность» и «без взятия» зависит от того, принимает ли экспедитор груз в свое физическое владение. [1, 2]
Согласно утвержденным ФНС и Минтрансу правилам транспортного ЭДО, ключевым маркером этого разделения является оформление сопутствующего документа — Электронной экспедиторской расписки (ЭЭР). [1, 2]

---

1. Перевозка БЕЗ взятия груза во владение (под ответственность)

В данном случае экспедитор выступает исключительно как агент (посредник). Он выполняет чисто организационную и интеллектуальную работу: подбирает транспорт, планирует маршрут, готовит документы или организует таможенное оформление, но сам к товару не прикасается. [1, 3, 4]

- Когда применяется: Клиент своими силами (или силами своих складских рабочих) отгружает товар напрямую в машину фактического перевозчика, которого нанял

экспедитор.
- Документооборот: Оформляется только ЭПЭ (задание на организацию логистики). Электронная экспедиторская расписка (ЭЭР) не создается, поскольку факта передачи товара экспедитору не было. В Электронной транспортной накладной (ЭТрН) в качестве грузоотправителя фигурирует сам клиент. [1, 2]
- Предел ответственности: Если груз по дороге украдут или разобьют, экспедитор по умолчанию не отвечает за действия перевозчика (если иное прямо не прописано в договоре). Он будет виноват, только если клиент докажет, что экспедитор проявил халатность при выборе перевозчика или допустил грубую ошибку в документах. [3, 4, 5]

2. Перевозка С ВЗЯТИЕМ груза во владение и под ответственность

Здесь экспедитор берет на себя роль договорного перевозчика (принципала) и полностью отвечает за сохранность груза с момента приемки до момента выдачи. [5, 6]

- Когда применяется: Сборные грузы (LTL), перевозки через кросс-докинг или консолидационные склады экспедитора, а также когда экспедитор забирает груз своим брендированным транспортом.
- Документооборот: Наряду с обязательным ЭПЭ экспедитор в обязательном порядке формирует Электронную экспедиторскую расписку (ЭЭР). Именно ЭЭР подтверждает, что экспедитор взял материальную ответственность за товар на себя. В ЭТрН в роли грузоотправителя часто указывается сам экспедитор. [1, 2, 7]
- Предел ответственности: Действует презумпция вины экспедитора. Если с грузом что-то случится в пути, экспедитор обязан выплатить клиенту полную стоимость ущерба, даже если фактически виноват сторонний водитель. Освободиться от ответственности экспедитор сможет, только если докажет форс-мажор. [5, 6, 8, 9]

Применение в субэкспедировании (цепочки экспедиторов)

Субэкспедирование — это схема, при которой Главный экспедитор (которого нанял Клиент) привлекает для выполнения задачи другого экспедитора (Субэкспедитора).
В электронном документообороте (ЭДО) это порождает каскадную цепочку документов, где один и тот же участник меняет роли:

```
[Клиент] --(ЭПЭ №1 / ЭЭР №1)--> [Главный экспедитор] --(ЭПЭ №2 / ЭЭР №2)--> [Субэкспедитор]
```

Сценарий А: Субэкспедирование БЕЗ взятия во владение

Если Главный экспедитор нанимает Субэкспедитора только для того, чтобы тот «нашел машину» через свои каналы:

1. Клиент отправляет ЭПЭ №1 Главному экспедитору.
2. Главный экспедитор перевыпускает задание и отправляет ЭПЭ №2 Субэкспедитору.
3. Расписки (ЭЭР) в этой цепочке вообще не выпускаются, так как никто из посредников груз физически не трогает. Товар загружается со склада Клиента сразу в машину конечного перевозчика. [1, 2]

Сценарий Б: Субэкспедирование С ВЗЯТИЕМ под ответственность

Самый частый сценарий в крупных логистических хабах и при мультимодальных перевозках:

1. Звено Клиент — Главный экспедитор: Клиент создает ЭПЭ №1. Как только Главный экспедитор забирает груз, он подписывает и отправляет Клиенту ЭЭР №1, подтверждая, что взял товар под свою материальную ответственность. [1, 2, 7]
2. Звено Главный экспедитор — Субэкспедитор: Для Субэкспедитора Главный экспедитор теперь сам становится «Клиентом». Он формирует ЭПЭ №2 (в рамках своего договора с субэкспедитором). При передаче груза на терминал или в машину Субэкспедитора, тот выпускает ЭЭР №2 на имя Главного экспедитора. [1, 2, 10]

> Важный юридический нюанс: При субэкспедировании с ответственностью Главный экспедитор полностью отвечает перед Клиентом по ЭЭР №1. Если Субэкспедитор потеряет груз, Клиент все равно потребует деньги с Главного экспедитора. И уже после этого Главный экспедитор в порядке регресса будет взыскивать убытки с Субэкспедитора, используя как доказательство ЭЭР №2. [6, 7]

Если вы планируете выстраивать цепочки субэкспедирования, подскажите: вы выступаете в роли главного экспедитора или субподрядчика? Могу детально разобрать, какие именно поля и титулы в ЭПЭ/ЭЭР вам необходимо контролировать для минимизации рисков.

[1] https://www.diadoc.ru

[2] https://zakupki-kontur.ru

[3] https://www.vantraa.nl

[4] https://ship4wd.com

[5] https://rapsinews.ru

[6] https://www.consultant.ru

[7] https://esphere.ru

[8] https://www.advgazeta.ru

[9] https://brace-lf.com

[10] https://pravo.ru

Мы выступаем от имени Главного экспедитора.
Производим мультимодальную контейнерную перевозку из Шанхая во Владивосток морем, Владивосток-Москва по ЖД, доставка до склада клиента на Авто.
Можем ли мы не брать груз в ответственность на весь маршрут, если мы субэкспедируем все 3 плеча

перевозки?

Нет, полностью снять с себя материальную ответственность перед клиентом на всем маршруте в такой схеме вы не можете.
Даже если вы субэкспедируете 100% маршрута (море + ЖД + авто) и физически не прикасаетесь к контейнеру, с точки зрения закона РФ вы остаетесь Главным экспедитором (Генеральным подрядчиком). [1]
Согласно ст. 805 ГК РФ и актуальной судебной практике Верховного Суда РФ, привлечение третьих лиц (субподрядчиков) не освобождает Главного экспедитора от ответственности перед Клиентом. Любые пункты в договоре в духе *«Экспедитор не отвечает за действия третьих лиц / порчу груза в пути»* признаются судами ничтожными, так как они прямо нарушают императивные нормы ФЗ № 87 «О транспортно-экспедиционной деятельности». [1, 2, 3]
Однако вы можете грамотно минимизировать риски и выстроить зеркальную систему претензий.

Как это работает на практике и где ваши риски

При мультимодальной перевозке цепочка выглядит так:
`Клиент` ➔ `Вы (Главный экспедитор)` ➔ `3 разных Субподрядчика (линия, РЖД/оператор, автоперевозчик)`
Если на каком-то из этапов контейнер вскроют, зальют водой или уронят, Клиент не будет разбираться, кто виноват. Он выставит претензию вам, и вы будете обязаны выплатить ему ущерб. [1, 4]
Но закон дает вам право регресса: выплатив деньги Клиенту, вы перевыставляете эту сумму тому субэкспедитору/перевозчику, на чьем плече произошло ЧП. [1]

Инструменты защиты для Главного экспедитора

Чтобы не остаться крайним при субэкспедировании, вам необходимо внедрить три уровня защиты:

1. Зеркальные договоры и жесткие ЭПЭ/ЭЭР с субподрядчиками

- С каждым субподрядчиком на каждом плече у вас должен быть подписан договор с принятием груза в ответственность (с выдачей вам Электронной экспедиторской расписки — ЭЭР или сквозного коносамента/накладной).
- Море (Шанхай — Владивосток): Субэкспедитор или морская линия обязаны выдать морской коносамент (Bill of Lading), где вы или ваш агент указаны в качестве получателя/нотифай-парти.
- ЖД (Владивосток — Москва): ЖД-экспедитор/оператор оформляет железнодорожную накладную (СМГС/ГУ-27).
- Авто (Москва — Склад): Автоперевозчик принимает ЭПЭ и подписывает ЭТрН (электронную транспортную накладную), где в Титуле 2 (Т2) водитель расписывается в приемке груза.

2. Коммерческие акты на стыках плеч (Критически важно!)

Самая большая проблема мультимодальных перевозок — определить, где именно повредили груз. Если контейнер приедет в Москву с сорванной пломбой или помятым боком, автоперевозчик скажет, что «так забрал с ЖД», ЖД скажет, что «так приехало из порта», а порт — что «такой сгрузили с судна».

- Обязуйте субподрядчиков проводить осмотр контейнера и сверку пломб на каждом стыке (Причал ➔ ЖД-тупик; ЖД-станция ➔ Автомашина).
- В случае малейших расхождений или повреждений (даже царапин на контейнере) субподрядчик обязан составить коммерческий акт (или акт общей формы) и прислать вам фото. Только так вы сможете доказать, какой именно субподрядчик отвечает за ущерб.

3. Страхование ответственности экспедитора

Поскольку судебные разбирательства с РЖД или морскими линиями могут длиться годами, а клиент потребует деньги сразу, оформите полис страхования ответственности экспедитора. При наступлении страхового случая (например, опрокидывание фуры на автоплече или повреждение на ЖД) страховая компания выплатит ущерб клиенту, а затем сама займется судами с виновным субподрядчиком. [5, 6]

Если вам необходимо, я могу подсказать, какие формулировки по переходу ответственности и фиксации повреждений на стыках лучше всего прописать в договоре с вашими субподрядчиками. Информацию по какому плечу (море, ЖД или авто) разобрать подробнее?

[1] https://rapsinews.ru

[2] https://www.consultant.ru

[3] https://13.rospotrebnadzor.ru

[4] https://www.consultant.ru

[5] https://docs.cntd.ru

[6] https://katsaylidi.ru

В каких сценариях применимо поручение экспедитору без принятия в ответственность?

Поручение экспедитору без принятия груза в ответственность применяется в сценариях, когда вы выступаете исключительно как информационный, юридический или организационный посредник (агент). [1, 2]
В логистике и ГК РФ этот формат официально называется «выполнение услуг без принятия груза во владение». Поручение экспедитору (ЭПЭ) в этом случае выполняет роль заявки-задания на услуги, но сопутствующая экспедиторская расписка (ЭЭР), запускающая материальную ответственность, не выпускается. [1, 2, 3]
Ниже приведены 4 основных практических сценария, когда Главный экспедитор оформляет ЭПЭ без принятия ответственности:

1. Чистое посредничество («Диспетчеризация»)

Вы нашли для клиента фактического перевозчика, согласовали с ним ставку и условия, но ваш персонал и техника к грузу не прикасаются.

- Как это работает: Клиент оформляет на вас Поручение (ЭПЭ). Вы одобряете его (выпускаете Титул 2), подтверждая, что обязуетесь организовать рейс. На погрузку приезжает машина нанятого вами перевозчика. Клиент сам передает груз напрямую водителю. [1, 2]
- Документооборот: Оформляется ЭПЭ + ЭТрН (Электронная транспортная накладная). В ЭТрН в Титуле 1 грузоотправителем значится ваш Клиент, а перевозчиком — фактический исполнитель (линия, автопарк). Вы в накладной можете быть указаны только в роли «Лица, организовавшего перевозку». Вашей подписи в графах приема/сдачи груза нет. [1, 4]

2. Консалтинг, таможенное оформление и документальное сопровождение

Клиент нанимает вас не для физического перемещения контейнера, а для преодоления юридических и бюрократических барьеров (например, на стыке плеч в порту Владивостока).

- Как это работает: В рамках мультимодального ЭПЭ вашей задачей является: подача деклараций, организация фитосанитарного/ветеринарного контроля, оплата портовых сборов, переоформление внутрипортовых документов.
- Документооборот: Ваша подпись стоит только под ЭПЭ и актом выполненных услуг. Контейнер все время находится на балансе и под ответственностью морской линии, а затем — терминала порта. [5]

3. Схема с прямой переадресацией ответственности (Прямой коносамент / накладная)

Этот сценарий часто используют, чтобы легально «обойти» материальную ответственность Главного экспедитора перед клиентом по ГК РФ.

- Как это работает: Клиент дает вам Поручение (ЭПЭ) организовать плечо (например, Шанхай — Владивосток). Вы нанимаете морскую линию, но просите её выписать Сквозной морской коносамент (или ЖД-накладную) напрямую на имя вашего Клиента в качестве грузополучателя, а не на вашу компанию.
- Результат: Вы выполнили поручение по организации, но юридически право владения и риски по транспортному документу перешли напрямую от линии к вашему клиенту.

4. Организация перевозок спецтехники, самоходных машин или выкупленных контейнеров (SOC)

Сценарий, когда товар перемещается «своим ходом» или на условиях, где сохранность обеспечивает специфика самого груза, а экспедитор лишь фрахтует место.

- Как это работает: Вы фрахтуете место на судне под личные порожние контейнеры клиента (SOC-контейнеры) или организуете перегон спецтехники под управлением водителей заказчика. Вы отвечаете по Поручению только за своевременное предоставление слота/разрешений, но не за сохранность имущества.

Главное правило безопасности для Главного экспедитора:

Если вы работаете по любому из этих сценариев и не хотите отвечать за повреждение груза, в вашем договоре транспортной экспедиции с клиентом НЕ должно быть фразы:

- *«Экспедитор принимает на себя обязательства по доставке груза и отвечает за его сохранность с момента принятия до момента выдачи»*.

Вместо этого предмет договора должен звучать так:

- *«Экспедитор обязуется за вознаграждение организовать выполнение услуг, привлекая третьих лиц от своего имени, но за счет Клиента, без принятия груза во владение Экспедитора»*. [2, 6]

Если вам требуется, мы можем составить формулировку для предмета договора, которая разграничит ваши обязанности по организации и освободит от рисков физической порчи груза. Написать такой пункт?

[1] https://kontur.ru

[2] https://event.kontur.ru

[3] https://www.diadoc.ru

[4] https://astral.ru

[5] https://www.diadoc.ru

[6] https://www.klerk.ru

В каких случаях Главный экспедитор не должен брать на себя ответственность за груз?

Главный экспедитор имеет полное право и прямые коммерческие показания не брать на себя материальную ответственность за груз в тех случаях, когда он физически и технологически не может контролировать его сохранность.
Ниже приведены основные случаи, когда вам, как Главному экспедитору, строго рекомендуется исключать пункт о принятии груза под свою ответственность в договоре и Поручении экспедитору (ЭПЭ):

1. Перевозка закрытых, опломбированных контейнеров (FCL) без вашего участия в погрузке

Если в Шанхае фабрика сама загружает товар в контейнер, вешает свою пломбу, а вы подключаетесь только на этапе фрахта судна.

- Почему нельзя брать ответственность за груз: Вы не знаете, что внутри, как товар закреплен и не побьется ли он о стенки контейнера при качке.
- Как действовать: Вы принимаете ответственность исключительно за целостность пломбы и внешнее состояние контейнера, но не за его содержимое. В ЭПЭ и договоре фиксируется: *«Прием груза осуществляется по наружному осмотру контейнера и номерам пломб отправителя»*.

2. Сверхдорогие, уникальные или высокорисковые грузы

Перевозка электроники, оборудования на миллионы долларов, фармацевтики с жестким температурным режимом или опасных грузов (ADR).

- Почему нельзя брать ответственность за груз: В случае ЧП (например, аварии на ЖД или поломки реф-контейнера) сумма убытков может мгновенно обанкротить вашу экспедиторскую компанию.
- Как действовать: Вы выступаете исключительно как организатор (агент). Обязанность по страхованию самого груза (Cargo Insurance) полностью возлагается на Клиента, либо вы страхуете его за счет клиента, но выгодоприобретателем в полисе указывается сам Клиент.

3. Использование контейнеров и транспорта самого Заказчика (SOC-

контейнеры)

Сценарий, когда клиент предоставляет свои собственные (или арендованные им напрямую) контейнеры, а вы лишь покупаете для них «место» на судне или ЖД-платформе.

- Почему нельзя брать ответственность за груз: Если контейнер окажется дырявым, и груз зальет дождем, или если у контейнера в пути оторвется балка из-за износа металла, это будет виной владельца оборудования (Клиента).
- Как действовать: В Поручении прописывается, что экспедитор отвечает только за предоставление слота/платформы, а техническое состояние транспортного оборудования обеспечивает Заказчик.

4. Сложные мультимодальные цепочки с «непрозрачными» государственными монополиями

В вашем маршруте присутствует ЖД-плечо (Владивосток — Москва). Взаимодействие с РЖД или крупными инфраструктурными терминалами часто идет по жестким внутренним уставам, где отсудить у них реальный ущерб крайне сложно.

- Почему нельзя брать ответственность за груз: Если вы возьмете груз под ответственность перед клиентом, то клиент взыщет с вас 100% ущерба по ГК РФ. А вот вы с РЖД по их Уставу сможете взыскать лишь копейки. Эта разница станет вашим чистым убытком.
- Как действовать: Договор составляется по агентской схеме. Вы обязуетесь организовать перевозку, а в случае повреждения груза на ЖД-плече обязуетесь передать Клиенту все права для прямой подачи претензии фактическому виновнику (ЖД-оператору).

5. Отсутствие технической возможности проверить груз при приемке

Если по маршруту Шанхай — Владивосток — Москва груз идет навалом или это сборный груз, но у вас нет своего склада во Владивостоке для его пересчета при перевалке с моря на ЖД.

- Почему нельзя брать ответственность за груз: Вы физически не можете проверить, не украли ли часть коробок в порту во время перевалки, пока груз не прибудет в Москву.
- Как действовать: Исключать выдачу Экспедиторской расписки. Использовать только Поручение без владения.

Как это правильно зафиксировать в документах:

Чтобы легально не брать ответственность в этих случаях, ваша связка документов должна выглядеть так:

1. В Договоре: Формулировка *«Экспедитор оказывает услуги без принятия груза в свое владение и ведение»*.
2. В ЭПЭ (Поручении): В поле «Вид экспедиторских услуг» четко писать: *«Организация перевозки (без принятия груза под ответственность Экспедитора)»*.
3. Экспедиторская расписка: Не оформляется ни в каком виде.

Если вы хотите подстраховать текущую перевозку Шанхай — Москва, напишите, застрахован ли сейчас этот груз самим клиентом? Могу подсказать, как правильно прописать в ЭПЭ оговорку о страховании, чтобы снять риски с вашей компании.

То есть не стоит принимать ответственность за груз на весь мультимодальный маршрут в первом ЭПЭ от Клиента к нам?

Абсолютно верно. Принимать на себя сквозную ответственность за весь мультимодальный маршрут в первом ЭПЭ от Клиента — это критический риск для Главного экспедитора.
Если вы подпишете ЭПЭ в таком формате, вы юридически превратитесь из организатора логистики в «страховую компанию» для своего клиента. Любое ЧП на море, железной дороге или автодороге станет вашей финансовой проблемой, даже если вы сработали идеально.
Чтобы обезопасить свой бизнес, вам нужно перестроить схему взаимодействия с клиентом.

Правильная стратегия работы с ЭПЭ и ответственностью

Вместо принятия сквозной ответственности на весь маршрут, используйте один из двух подходов:

Подход 1. Оформление ЭПЭ БЕЗ принятия ответственности (Рекомендуемый)

Это лучший вариант для Главного экспедитора в мультимодальных перевозках.

1. Что делать в ЭПЭ: В Титуле 2 (вашем согласии на Поручение) вы подтверждаете только организацию процесса.
2. Экспедиторская расписка (ЭЭР): Вы её не выпускаете вообще.
3. Как распределяется ответственность: По договору и ЭПЭ вы отвечаете только за убытки, вызванные вашими личными ошибками (например, неверно заполнили таможенную декларацию, из-за чего контейнер застрял и клиент попал на демередж). За физическую сохранность груза отвечают конкретные фактические перевозчики (морская линия, РЖД-оператор, автопарк), каждый на своем плече.

Подход 2. Дробление ответственности по плечам (Если клиент настаивает)

Если крупный клиент ставит жесткое условие: *«Либо вы берете груз под ответственность, либо мы уходим к конкурентам»*, ни в коем случае не подписывайте сквозное поручение на весь маршрут (Шанхай — Владивосток — Москва — Склад). Разбейте процесс на этапы:

- Плечо 1 (Море: Шанхай — Владивосток): В ЭПЭ фиксируется: *«Без принятия во владение. Ответственность за сохранность несет морская линия согласно коносаменту»*.
- Плечо 2 (ЖД: Владивосток — Москва): Ответственность также не принимается, так как вы не можете контролировать сохранность контейнера на путях РЖД.
- Плечо 3 (Авто: Москва — Склад клиента): Только на этом последнем этапе, если ваш привлеченный водитель забирает контейнер с ЖД-станции, вы можете выпустить ЭПЭ/ЭТрН *с ответственностью*, но она будет действовать строго с момента погрузки на авто до ворот склада клиента. При этом риск автоплеча вы мгновенно

«зеркалите» на автоперевозчика.

Что вам нужно сделать прямо сейчас в документах:

1. Проверить предмет договора с Клиентом. Убедитесь, что там написано: *«Экспедитор обязуется за вознаграждение и за счет Клиента организовать выполнение услуг...»*, а не *«...обязуется доставить груз»*. Фраза «организовать услуги» — ваш главный юридический щит.
2. Внести оговорку в ЭПЭ. В строке «Особые отметки» или «Условия перевозки» электронного поручения обязательно пропишите:

> *«Перевозка осуществляется без принятия груза во владение Экспедитора. Ответственность за сохранность груза распределяется между фактическими перевозчиками на каждом этапе транспортировки в соответствии с международными конвенциями и уставами видов транспорта»*.

Если клиент категорически требует сквозную ответственность и не соглашается на другие условия, единственным выходом будет включение стоимости полиса страхования груза (Cargo) в ваш тариф.
Хотите, я помогу сформулировать официальный и аргументированный ответ для вашего клиента, почему в мультимодальных перевозках ЭПЭ оформляется без сквозного принятия ответственности?

ФСБ

Постановление № 935 от 01.11.2023: Разбор технических требований и архитектуры СОРМ

Важное уточнение

По указанной дате (01.11.2023) под номером 935 был принят не правительственный документ, а **Приказ Минцифры России № 935**. Данный нормативно-правовой акт утверждает **требования к вычислительной мощности хостинг-провайдеров для внедрения СОРМ** (систем оперативно-разыскных мероприятий).

Если вы искали именно Постановление Правительства РФ, регулирующее эту же сферу (деятельность хостеров и передачу данных ФСБ), то аналогичные нормы зафиксированы в **Постановлении Правительства РФ № 1952 от 22.11.2023**.

Главное о Приказе Минцифры № 935 от 01.11.2023

Документ детально описывает технические правила взаимодействия между ИТ-системами хостинг-провайдеров и спецслужбами для включения компаний в реестр Роскомнадзора:

- **Технические требования**: Хостеры обязаны предоставить ФСБ удаленный доступ к базам данных и вычислительным мощностям.

- **Протоколы связи**: Для обмена командами и выгрузки информации строго прописано использование архитектуры данных **GraphQL**.
- **Собираемые данные**: Определен жесткий перечень логов, сетевых адресов и регистрационных данных пользователей, которые хостер должен хранить и автоматически передавать по запросу ведомств.

Таблица смежных нормативных актов конца 2023 года

Деятельность хостинг-провайдеров регулируется целым пакетом документов, принятых в этот период:

Документ	Основное содержание
Приказ Минцифры № 935 (от 01.11.2023)	Технические требования к базам данных хостеров под нужды СОРМ
Постановление Правительства № 1952 (от 22.11.2023)	Общие правила взаимодействия провайдеров со спецслужбами
Постановление Правительства № 2008 (от 28.11.2023)	Порядок ведения и вхождения компаний в официальный реестр РКН

Пример предоставления данных по регламентам этого постановления

Согласно **Приказу Минцифры № 935 от 01.11.2023**, обмен данными между ИС хостинг-провайдера и техническими средствами ОРМ (спецслужбами) осуществляется автоматически через **язык запросов GraphQL** по протоколу **WebSocket**. Провайдер не передает файлы вручную, а разворачивает API, к которому удаленно обращается система СОРМ.

Ниже приведен абстрактный технический пример того, как выглядит типичный запрос спецслужб к базе хостера и структурированный ответ на него в соответствии с

регламентом.

Сценарий: Запрос данных клиента по IP-адресу

Сотрудник ведомства через пульт управления СОРМ хочет узнать, кому принадлежит виртуальный сервер (VPS), с которого фиксировалась активность под конкретным IP-адресом.

---

1. Исходящий запрос от СОРМ (GraphQL-запрос)

Информационная система ведомства отправляет стандартизированный запрос, где указывает критерий поиска (например, целевой `ipAddress` и временной отрезок):

graphql

```
query GetAbonentByIP {
  # Запрос информации о пользователе ИС хостинга
  findUsersByNetworkActivity(
    ipAddress: "198.51.100.42"
    timestamp: "2026-07-09T12:00:00Z"
  ) {
    id
    registrationDate
    contractNumber
    status
    # Тип пользователя: физическое лицо
    personProfile {
      fullName
      birthDate
      passportSeries
      passportNumber
    }
    # Данные об используемой вычислительной мощности (сервисе)
    activeServices {
      serviceType
      allocatedIpAddresses
      domainNames
    }
  }
}
```

---

2. Автоматический ответ хостинг-провайдера (GraphQL-ответ)

Вычислительные мощности провайдера (специальный модуль COPM) обрабатывают этот запрос, обращаются к внутренней биллинговой системе и логам, а затем возвращают строго структурированный JSON-ответ:

json


```
{
  "data": {
    "findUsersByNetworkActivity": [
      {
        "id": "usr_948201",
        "registrationDate": "2024-03-15",
        "contractNumber": "XCT-2024/77",
        "status": "ACTIVE",
        "personProfile": {
          "fullName": "Иванов Петр Сергеевич",
          "birthDate": "1990-05-24",
          "passportSeries": "4512",
          "passportNumber": "654321"
        },
        "activeServices": {
          "serviceType": "VPS_HOSTING",
          "allocatedIpAddresses": ["198.51.100.42"],
          "domainNames": ["example-target-site.ru"]
        }
      }
    ]
  }
}
```

Какие еще данные предоставляются по схемам Приказа № 935:

- **Логи авторизации**: Информация о дате, времени и IP-адресах, с которых сам клиент заходил в личный кабинет хостинга или по SSH/FTP на свой сервер.
- **Платежные данные**: Номера банковских карт, кошельков или счетов, с которых производилась оплата услуг хостинга.
- **История изменений**: Сведения о смене DNS-записей, привязке новых доменов или изменении конфигурации арендуемых серверов.

Если вам нужны **конкретные схемы типов (Type Definitions)** из приложений к приказу или технические условия по организации каналов связи, уточните задачу.

Есть ли в этом регламенте требования к полям ответов?

Да, в Приказе Минцифры № 935 зафиксированы жесткие и исчерпывающие требования к полям ответов. Документ содержит приложения, которые представляют собой готовую **техническую спецификацию (схему GraphQL)**. Провайдер хостинга не имеет права менять названия полей, их типы или структуру по своему усмотрению.

Все поля ответов делятся на строго регламентированные категории:

1. Требования к типам данных (Data Types)

Каждое поле в ответе должно строго соответствовать базовым типам GraphQL или специальным скалярам:

- **Идентификаторы**: Поля типа `id` должны иметь тип `ID!` (уникальная строка, обязательная к заполнению).
- **Временные метки**: Все поля времени (регистрация, авторизация, логи) должны передаваться в формате ISO 8601 (строгий тип `DateTime`), например: `2026-07-09T12:00:00Z`.
- **Сетевые параметры**: Для IP-адресов регламент требует поддержки типов данных под IPv4 и IPv6.

2. Обязательность заполнения полей (Non-Null Fields)

В схеме приказа множество полей отмечены знаком `!`. Это значит, что если уполномоченный орган запрашивает запись, хостер **обязан** вернуть значение в этих полях. Возврат значения `null` (пустоты) по ним считается нарушением регламента. К обязательным полям относятся:

- Внутренний ID пользователя в системе хостинга.
- Дата заключения договора/создания аккаунта.
- Статус учетной записи (активен, заблокирован).

3. Строгая структура вложенности (Объекты)

Регламент запрещает отдавать данные «плоским» текстом. Ответ должен быть строго нормализован:

- Данные о физическом лице должны быть упакованы исключительно в объект профиля (например, `personProfile`) со строго фиксированными дочерними полями: `fullName`, `passportSeries`, `passportNumber`.
- Данные о юридическом лице — в объект организации с полями `inn`, `ogrn`, `companyName`.

4. Использование жестких перечислений (Enum)

Для многих полей ответов регламент устанавливает закрытый список возможных текстовых значений. Хостер не может написать статус «в бане» или «отключен за неуплату». Система должна возвращать строго определенные в API маркеры, например:

- Для статуса аккаунта: `ACTIVE`, `SUSPENDED`, `TERMINATED`.
- Для типа сервиса: `VPS`, `DEDICATED`, `SHARED_HOSTING`.

Если вам необходимо узнать, как в регламенте описываются поля для **конкретного типа данных** (например, платежных транзакций или логов подключения к панели управления), напишите, и мы разберем эти структуры.

Сколько вариантов запросов может поступить по этому регламенту?

В рамках технического регламента Приказа Минцифры № 935 от 01.11.2023 на сервер хостинг-провайдера может поступить **около 20 основных базовых типов GraphQL-запросов (Query)**. Поскольку архитектура GraphQL гибка и позволяет комбинировать аргументы, количество *комбинаций* запросов исчисляется сотнями, но сам жестко зафиксированный перечень целевых поисковых функций (методов) ограничен.

Все регламентированные типы запросов от пульта ФСБ делятся на **5 ключевых категорий**, соответствующих приложениям к приказу:

1. Запросы по идентификации клиентов (Абонентов)

Это базовые поисковые функции для сопоставления виртуальных ресурсов с реальными личностями или компаниями. Спецслужбы могут запрашивать:

- Поиск пользователя по **ФИО** / наименованию организации.
- Поиск по паспортным данным или **ИНН** / ОГРН.
- Поиск по контактным данным (номер телефона, e-mail).
- Поиск по **номеру договора** или уникальному ID в биллинге хостера.

2. Запросы по сетевым идентификаторам и ресурсам

Позволяют установить владельца цифрового следа. Поступают в случаях, когда известен только адрес в сети:

- Поиск по **IP-адресу** (выделенному под VPS, выделенный сервер или shared-хостинг) на определенный момент времени.
- Поиск по **доменному имени**, размещенному на серверах провайдера.
- Поиск по MAC-адресу или внутреннему сетевому интерфейсу.

3. Запросы истории авторизаций и сессий управления

Применяются для отслеживания действий администраторов сайтов:

- Запрос логов входа в **личный кабинет** хостинга.
- Запрос истории подключений к панелям управления (cPanel, ISPmanager и др.).
- Запрос сессий удаленного доступа по протоколам **SSH, FTP, RDP** к арендуемым вычислительным мощностям.

4. Финансовые и транзакционные запросы

Служат для установления источников финансирования интернет-ресурсов:

- Запрос истории **платежей** пользователя за услуги хостинга.
- Поиск клиента по номеру банковской карты или электронного кошелька, с которых поступала оплата.
- Запрос детализации по конкретной банковской транзакции.

5. Служебные (системные) запросы

Используются пультом управления СОРМ для проверки работоспособности самого модуля интеграции:

- Запрос на проверку связи (**Ping / Heartbeat**).
- Запрос актуальной схемы данных (`getSchema`) для подтверждения соответствия версии API.
- Запросы статистики и объемов хранящихся логов.

Как обычно реализовывают обмен по этому постановлению?

Архитектура обмена данными строится именно по принципу предоставления API со стороны наблюдаемого хостинга для оборудования ФСБ. На техническом сленге интеграторов эта схема называется ИС СОРМ (Информационная Система СОРМ).

Однако, вопреки классическому веб-развитию, это **не обычный HTTP API с GET/POST-запросами**, а гибридная схема на базе **WebSockets и HTTP**, где в качестве языка общения строго закреплен **GraphQL**, а данные упакованы в формат **JSON**.

Как это устроено технически: пошаговая схема

1. **Хостер разворачивает ИС (базу данных и API)**: Провайдер своими силами (или покупая готовое ПО у сертифицированных СОРМ-интеграторов, например, «Ящер-935») разворачивает специальный программный модуль. Этот модуль имеет прямой доступ к биллингу, логам серверов, базам авторизации и персональным данным.
2. **Оборудование ФСБ подключается к хостеру**: На стороне хостинга организуется выделенный, защищенный шифрованием (часто по ГОСТ) канал связи. Пульт

управления ОРМ (оборудование спецслужб) выступает в роли **клиента**, который подключается к API хостинга.
3. **Запросы поступают от ФСБ к хостеру**: Сотрудник ведомства через свой интерфейс отправляет запрос. Пульт ФСБ генерирует GraphQL-запрос и отправляет его в ИС хостинга. Провайдер обрабатывает его «на лету» и возвращает ответ в JSON.

Почему это НЕ обычный HTTP (GET/POST)?

Хотя GraphQL технически можно гонять через стандартные `POST`-запросы (и в некоторых служебных методах приказа HTTP используется), **основным транспортом для оперативного взаимодействия по Приказу № 935 является протокол WebSocket**.

Критерий	Как в обычном Web	Как реализовано в СОРМ (Приказ № 935)
Протокол (Транспорт)	HTTP / HTTPS (постоянное переоткрытие соединений)	**WebSocket** (устанавливается одно постоянное полнодуплексное соединение)
Методы	GET / POST / PUT / DELETE	**GraphQL операции**: `Query` (запросы) и `Subscription` (подписки на события в реальном времени)
Формат данных	Любой (JSON, XML, Form-Data)	Строго **JSON** (согласно синтаксису GraphQL-ответов)

Зачем спецслужбам понадобился WebSocket и GraphQL?

- **Подписки (Subscriptions)**: Обычный HTTP API требует постоянного опроса (polling) сервера: *«Появились новые логи? А сейчас? А сейчас?»*. WebSocket позволяет пульту ФСБ один раз подписаться на событие. Как только злоумышленник авторизуется в панели хостинга, сервер хостера **сам моментально проталкивает (push)** JSON-пакет в сторону ведомства.
- **Экономия трафика и гибкость**: GraphQL позволяет пульту ФСБ запрашивать ровно те поля, которые нужны в данный момент. Если оператору нужен только ИНН компании, сервер хостера не будет собирать и упаковывать тяжелые логи её серверов. Это снижает нагрузку на вычислительные мощности провайдера.

Резюме для разработчика

Чтобы соответствовать регламенту, вам нужно написать или внедрить приложение, которое:

- Открывает защищенный порт наружу (только для IP-адресов пульта ФСБ).
- Поднимает **WebSocket-сервер**.
- Умеет парсить входящий **GraphQL-синтаксис** (валидировать его по жесткой схеме из Приказа Минцифры).
- Отдает структурированный **JSON** в рамках открытой WebSocket-сессии.

Если вы планируете реализовывать это самостоятельно, уточните: **какой стек технологий или язык программирования** вы используете в биллинге? Я подкажу, какие готовые библиотеки для работы с GraphQL/WebSockets ускорят прохождение проверок.

Используем 1С. В последних версиях платформы присутствуют изменения по WebSockets

Использование встроенных механизмов **платформы «1С:Предприятие» (начиная с версии 8.3.27)** значительно упрощает интеграцию, однако здесь есть важнейший архитектурный нюанс.

Платформа 1С умеет выступать только в роли WebSocket-клиента, а не сервера. В дереве метаданных и встроенном языке появились объекты `Общие -> WebSocket-клиенты`, `WebSocketКлиентСоединения` и `WebSocketКлиенты`. Но по Приказу Минцифры № 935 **хостинг обязан поднять именно WebSocket-сервер**, к которому пульт ФСБ будет подключаться в качестве клиента.

Прямо «из коробки» запустить прослушивание WebSocket-порта средствами сервера 1С не получится. Чтобы решить эту задачу, используют одну из двух проверенных архитектурных схем.

Схема 1. Использование прокси-сервера (Рекомендуемая)

Поскольку 1С идеально умеет работать со стандартными **HTTP-сервисами**, между пультом ФСБ и 1С ставится промежуточный легковесный веб-сервер.

1. **Транспортный узел (Nginx / Node.js / Go)**: Снаружи открыт порт, который слушает WebSocket-подключения от спецслужб. Он берет на себя авторизацию, удержание постоянного соединения (keep-alive) и шифрование.
2. **Преобразование**: Когда от пульта ФСБ по WebSockets прилетает текстовый GraphQL-запрос, прокси-сервер пересылает его в 1С через обычный внутренний `POST`-запрос на стандартный HTTP-сервис 1С.
3. **Обработка в 1С**: 1С принимает GraphQL (строку), парсит JSON, запрашивает данные из базы, формирует ответный JSON и отдает его HTTP-сервису. Прокси-сервер перенаправляет этот JSON обратно в WebSocket-канал.

Плюсы: Безопасно, не нагружает сеансы 1С удержанием «пустых» соединений, легко масштабируется.

Схема 2. Использование Системы взаимодействия (Enterprise-подход)

Если в вашей инфраструктуре развернут **Сервер системы взаимодействия 1С** (1С:Управление конференциями / 1С:Диалог), задачу можно решить стандартными компонентами.

- Сервер системы взаимодействия внутри себя работает именно на WebSockets и Java.
- Вы можете опубликовать внешние точки подключения (через Webhooks или интеграционные API Системы взаимодействия).

Однако этот способ сложнее адаптировать под жесткие требования к JSON-структурам ответов GraphQL, прописанным в приказе.

Как реализовать GraphQL-логику внутри 1С?

Вне зависимости от того, как WebSocket-запрос попадет в 1С, вам придется обрабатывать спецификацию GraphQL средствами встроенного языка. Для этого используются три

механизма платформы:

1. Чтение и Запись JSON

GraphQL-запрос приходит в теле HTTP-запроса в виде структуры `{"query": "..."}`. Для разбора используется стандартный объект платформы:

bsl

```
Чтение = Новый ЧтениеJSON;
Чтение.УстановитьСтроку(СтрокаЗапросаОтФСБ);
ДанныеЗапроса = ПрочитатьJSON(Чтение);
ТекстGraphQL = ДанныеЗапроса.query; // Здесь будет сам текст GraphQL
```

2. Парсинг GraphQL (Поисковых полей)

Платформа 1С не имеет встроенного валидатора схем GraphQL. На практике разработчики пишут простой парсер строки (через `СтрНайти` / `СтрРазделить` или регулярные выражения через `RegExp`), чтобы вытащить:

- Название метода (например, `findUsersByNetworkActivity`)
- Параметры поиска (`ipAddress`, `timestamp`)

3. Формирование строгого ответа

Чтобы вернуть ответ в точном соответствии с регламентом (с учетом обязательных полей и типов данных), структура собирается через `Соответствие` или `Структура` и сериализуется в JSON:

bsl

```
// Формируем профиль физлица строго по схеме приказа
ПрофильФизЛица = Новый Структура;
ПрофильФизЛица.Вставить("fullName",        Выборка.ФИО);
ПрофильФизЛица.Вставить("passportSeries",  Выборка.СерияПаспорта);
ПрофильФизЛица.Вставить("passportNumber",  Выборка.НомерПаспорта);

// Собираем итоговый пакет
ДанныеАбонента = Новый Структура;
ДанныеАбонента.Вставить("id",            Строка(Выборка.Ссылка.УникальныйИдентификатор()));
ДанныеАбонента.Вставить("contractNumber", Выборка.НомерДоговора);
ДанныеАбонента.Вставить("personProfile",  ПрофильФизЛица);

// Оборачиваем в массив, как требует GraphQL
РезультатПоиска = Новый Массив;
РезультатПоиска.Добавить(ДанныеАбонента);

ОтветGraphQL = Новый Структура;
ОтветGraphQL.Вставить("data", Новый Структура("findUsersByNetworkActivity", РезультатПоиска));

// Записываем в JSON
```

```
ЗаписьJSON = Новый ЗаписьJSON;
ЗаписьJSON.УстановитьСтроку();
ЗаписатьJSON(ЗаписьJSON, ОтветGraphQL);
СтрокаОтветаJSON = ЗаписьJSON.Закрыть();
```

Если вы склоняетесь к архитектуре с **Nginx / Node.js проксированием** перед HTTP-сервисами 1С, напишите, и мы детальнее разберем, как **настроить маршрутизацию** или обрабатывать **Subscriptions (подписки на события)**, когда 1С должна сама инициировать отправку логов при их появлении.

Мне нравится схема 1. Покажи пример конфигурации nginx под него

Для реализации первой схемы **Nginx** подходит идеально. Он будет выступать в роли веб-сервера, который принимает постоянное WebSocket-соединение от пульта ФСБ, «приземляет» его и пересылает запросы на стандартный **HTTP-сервис 1С** (публикацию вашей информационной базы).

Поскольку в GraphQL-запросах по WebSockets обмен идет текстовыми сообщениями (фреймами), обычному Nginx для полноценной конвертации WebSocket-сообщений в HTTP-запросы `POST` к 1С требуется дополнительный скрипт-прослойка (например, микросервис на Node.js или Python).

Ниже приведен готовый пример архитектуры, где **Nginx** принимает трафик, защищает его, а легковесный скрипт **Node.js** передает JSON-данные в **1С**.

Архитектурная цепочка данных:

`Пульт ФСБ (WebSocket)` ➡ `Nginx (Порт 443)` ➡ `Node.js прокси (Порт 3000)` ➡ `HTTP-сервис 1С (Порт 80/443)`

---

1. Конфигурация Nginx (`nginx.conf`)

Этот конфиг решает главные задачи: включает шифрование, разрешает только определенные IP-адреса ведомства и проксирует WebSocket-протокол.

nginx

```
# Ограничение по IP: разрешаем доступ только пульту ФСБ
geo $permitted_fsb {
    default        0;
    195.178.0.0/24 1; # Замените на реальные IP-адреса пульта ведомства
    127.0.0.1      1; # Для локальных тестов
}

server {
    listen 443 ssl;
    server_name sorm.your-hosting.ru;

    # Блокируем всех, кто не входит в белый список IP
    if ($permitted_fsb = 0) {
        return 403;
    }

    # Настройки SSL/TLS (Требование безопасности СОРМ)
    ssl_certificate     /etc/ssl/certs/sorm_hosting.crt;
    ssl_certificate_key /etc/ssl/certs/sorm_hosting.key;
    ssl_protocols       TLSv1.2 TLSv1.3;
    ssl_ciphers         HIGH:!aNULL:!MD5;

    # Точка входа для WebSocket (Пульт ФСБ подключается сюда)
    location /graphql-ws {
        # Перенаправляем трафик на внутренний Node.js прокси
        proxy_pass http://127.0.0.1:3000;

        # Магия Nginx для поддержки WebSockets (Upgrade заголовки)
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "Upgrade";

        # Передаем реальный IP-адрес пульта вглубь системы
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

        # Таймауты для удержания постоянного соединения
        proxy_read_timeout 86400s;
        proxy_send_timeout 86400s;
    }
}
```

---

2. Скрипт-прослойка Node.js (`proxy.js`)

Так как Nginx сам по себе не умеет парсить внутренности WebSocket-сообщений и вызывать HTTP-сервисы, этот крошечный скрипт (всего около 30 строк кода) принимает текстовый GraphQL-пакет из WebSocket и делает обычный `POST`-запрос в вашу 1С.

Для его работы нужно установить две библиотеки: `npm install ws axios`.

javascript

```
const { WebSocketServer } = require('ws');
const axios = require('axios');

// Запускаем WebSocket сервер на внутреннем порту 3000
const wss = new WebSocketServer({ port: 3000 });

wss.on('connection', function connection(ws) {
    console.log('Пульт ФСБ подключился к WebSocket');

    // Слушаем входящие сообщения (GraphQL запросы) от пульта
    ws.on('message', async function message(data) {
        try {
            // Преобразуем входящий текстовый фрейм в читаемый вид
            const incomingQuery = data.toString();

            // Пересылаем запрос в HTTP-сервис 1С
            const responseFrom1C = await axios.post(
                'http://1c-server/base/hs/sorm_api/v1/query', // URL публикации вашего HTTP-серв
                { query: incomingQuery },
                {
                    headers: { 'Content-Type': 'application/json' },
                    auth: { username: 'SormUser', password: 'HardPassword1C' } // Авторизация в
                }
            );

            // Возвращаем JSON-ответ из 1С обратно в WebSocket-канал пульта ФСБ
            ws.send(JSON.stringify(responseFrom1C.data));

        } catch (error) {
            console.error('Ошибка обработки запроса:', error.message);
            ws.send(JSON.stringify({ errors: [{ message: "Внутренняя ошибка ИС хостинга" }] }));
        }
    });
});
```

Что нужно сделать в самой 1С?

В вашей конфигурации 1С создается стандартный **HTTP-сервис** (например, с именем `sorm_api`):

1. В нем объявляется шаблон URL: `v1/query`.
2. Метод: `POST`.
3. В обработчике метода вы пишете:

bsl

```
Функция v1queryPOST(Запрос)
    // Читаем входящий GraphQL-запрос, переданный из Node.js
    ТелоСтрокой = Запрос.ПолучитьТелоКакСтроку();

    // ... Ваша логика парсинга, поиска в БД и формирования структуры ответа ...

    // Возвращаем результат
    Ответ = Новый HTTPСервисОтвет(200);
    Ответ.УстановитьТелоИзСтроки(СтрокаJSONОтвета, КодировкаТекста.UTF8, ИспользованиеByteOrderM
    Возврат Ответ;
КонецФункции
```

Если вы хотите протестировать эту связку прямо сейчас на тестовом контуре, подсказать, как настроить **эмуляцию пульта ФСБ (клиента)** для проверки отправки запросов, или разобрать логику авторизации по ГОСТ-сертификатам?

Какой наиболее легковесный вариант вместо Node.js можно взять? Можно ли решить задачу с помощью Lua?

Да, вариант с Lua — это самое легковесное, быстрое и элегантное решение. Вам даже не придется запускать отдельный фоновый процесс (как Node.js), так как Lua встраивается прямо внутрь Nginx с помощью модуля **OpenResty** или `ngx_http_lua_module`.

В такой архитектуре Nginx сам превращается в полноценный WebSocket-сервер, парсит пакеты с помощью Lua и мгновенно перенаправляет их в HTTP-сервис 1С. Потребление оперативной памяти снизится со ~150 МБ (для Node.js) до **нескольких мегабайт**.

Архитектурная цепочка данных:

`Пульт ФСБ (WebSocket)` ➡ `Nginx + Lua (Порт 443)` ➡ `HTTP-сервис 1С (Внутренний POST-запрос)`

Инструкция по реализации на Nginx + Lua

Для работы этой схемы вам понадобится сборка Nginx с поддержкой Lua. В современных дистрибутивах Linux (Ubuntu/Debian) достаточно установить пакет **OpenResty** или стандартные пакеты `nginx` и `libnginx-mod-http-lua`. Также потребуется стандартная Lua-библиотека для работы с WebSockets (`lua-resty-websocket`).

1. Конфигурация Nginx со встроенным Lua-скриптом

Вся логика проксирования и подмены протоколов описывается прямо внутри блока `location` вашего `nginx.conf`:

nginx

```
server {
    listen 443 ssl;
    server_name sorm.your-hosting.ru;

    ssl_certificate     /etc/ssl/certs/sorm_hosting.crt;
    ssl_certificate_key /etc/ssl/certs/sorm_hosting.key;

    # Точка входа для пульта ФСБ
    location /graphql-ws {
        # Указываем Nginx, где искать DNS для внутренних HTTP-запросов к 1С
        resolver 8.8.8.8;

        # Вызов встроенного Lua-скрипта
        content_by_lua_block {
            -- Подключаем библиотеки OpenResty для работы с WebSocket и HTTP
            local server = require "resty.websocket.server"
            local http = require "resty.http"
            local cjson = require "cjson"

            -- Инициализируем WebSocket-соединение с пультом
            local wb, err = server:new{
                timeout = 86400000, -- Таймаут удержания соединения (в мс)
                max_payload_len = 65535
            }
            if not wb then
                ngx.log(ngx.ERR, "Ошибка инициализации WS: ", err)
                return ngx.exit(444)
            end

            -- Запускаем бесконечный цикл ожидания запросов от пульта ФСБ
            while true do
                local data, typ, err = wb:recv_frame()

                -- Если пульт закрыл соединение или произошла ошибка — выходим
                if not data or typ == "close" then
```

```
                    break
                end

                -- Обрабатываем только текстовые фреймы (GraphQL текстовые запросы)
                if typ == "text" then
                    -- Создаем HTTP-клиент для отправки запроса в 1С
                    local httpc = http.new()

                    -- Формируем JSON-пакет для 1С, упаковывая туда входящий GraphQL
                    local payload = cjson.encode({ query = data })

                    -- Делаем синхронный POST-запрос к HTTP-сервису 1С
                    local res, http_err = httpc:request_uri("http://1c-server/base/hs/sorm_api/v
                        method = "POST",
                        body = payload,
                        headers = {
                            ["Content-Type"] = "application/json",
                            ["Authorization"] = "Basic U29ybVVzZXI6SGFyZFBhc3N3b3JkMUM=" -- Base
                        }
                    })

                    -- Отправляем ответ из 1С обратно в WebSocket-канал пульта ФСБ
                    if res and res.status == 200 then
                        wb:send_text(res.body)
                    else
                        ngx.log(ngx.ERR, "1С вернула ошибку: ", http_err)
                        wb:send_text(cjson.encode({ errors = { { message = "Ошибка ИС хостинга"
                    end
                end
            end

            wb:send_close()
        }
    }
}
```

Почему решение на Lua идеальное для СОРМ?

1. **Минимальный след в системе**: Скрипт выполняется в контексте рабочих процессов (workers) самого Nginx. Ему не нужны Node.js-рантаймы или сторонние менеджеры процессов вроде PM2.
2. **Максимальное быстродействие**: Архитектура LuaJIT (Just-In-Time компилятор в OpenResty) по скорости обработки ввода-вывода практически не уступает чистому Си. Проксирование происходит с субмиллисекундными задержками.
3. **Безопасность**: Чем меньше сторонних пакетов и зависимостей (`node_modules`) крутится на сервере, тем меньше потенциальных уязвимостей в контуре СОРМ, что

облегчает сдачу узла связи контролирующим органам.

Если вы решите внедрять OpenResty с Lua, подсказать, как правильно **сгенерировать строку авторизации (Base64)** для заголовков 1С или как организовать **ротацию логов ошибок** самого Nginx, чтобы они не переполнили диск при частых запросах?

В ответах искусственного интеллекта могут быть ошибки. Если вам требуется юридическая консультация, обратитесь к специалисту.