ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 58123— 2018 (ИСО 15118-2: 2014)

ТРАНСПОРТ ДОРОЖНЫЙ

Интерфейс связи автомобиль — электрическая сеть

Часть 2

Требования к протоколу сетевого и прикладного уровней

(ISO 15118-2:2014, MOD)

Издание официальное

Москва
Стандартинформ
2018

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Федеральным государственным унитарным предприятием «Центральный ордена Трудового Красного Знамени научно-исследовательский автомобильный и автомоторный институт» (ФГУП «НАМИ») на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 56 «Дорожный транспорт»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 2 октября 2018 г. № 688-ст

4 Настоящий стандарт является модифицированным по отношению к международному стандарту ИСО 15118-2:2014 «Дорожные транспортные средства. Интерфейс связи транспортного средства и электросети. Часть 2. Требования к информационной сети и прикладному протоколу» (ISO 15118-2:2014 «Road vehicles — Vehicle-to-Grid Communication Interface — Part 2: Network and application protocol requirements», MOD) путем изменения отдельных фраз, слов, ссылок, которые выделены в тексте курсивом, а также путем изменения его структуры для приведения в соответствие с правилами, приведенными в ГОСТ 1.5—2001 (пункт 3.12).

Внесение указанных технических отклонений направлено на учет особенностей объекта стандартизации, характерных для Российской Федерации, и целесообразности использования ссылочных национальных стандартов вместо ссылочных международных стандартов.

Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте, приведены в дополнительном приложении ДА.

Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта приведено в дополнительном приложении ДБ

5 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.gost.ru)

© ISO, 2014 — Все права сохраняются
© Стандартинформ, оформление, 2018

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения . . . 1
2 Нормативные ссылки . . . 1
3 Термины и определения . . . 2
4 Обозначения и сокращения . . . 4
5 Понятия . . . 5
5.1 Определение сервисов на базе стандарта взаимодействия открытых систем (OSI) . . . 5
5.2 Структура требований . . . 5
5.3 Использование RFC-ссылок . . . 5
5.4 Представление, используемое для диаграмм XML-схем . . . 5
6 Обзор документов . . . 6
7 Базовые требования к связи V2G . . . 7
7.1 Общая информация . . . 7
7.2 Концепция сервисного примитива уровневой архитектуры OSI . . . 7
7.3 Концепция безопасности . . . 8
7.4 Состояния связи V2G и работа с каналом данных . . . 15
7.5 Канальный уровень . . . 20
7.6 Сетевой уровень . . . 20
7.7 Транспортный уровень . . . 21
7.8 Коммуникационный протокол V2G . . . 25
7.9 Представительский уровень . . . 28
7.10 Уровень приложения . . . 37
8 Сообщения прикладного уровня . . . 43
8.1 Общая информация и определения . . . 43
8.2 Определение подтверждения протокола . . . 44
8.3 Определение сообщения V2G . . . 47
8.4 Определения сеанса связи V2G и элементов тела сообщения . . . 49
8.5 Комплексные типы данных . . . 86
8.6 Идентификационные режимы и определения наборов сообщений . . . 119
8.7 Хронирование связи V2G . . . 156
8.8 Задание последовательности сообщений и обработка ошибок . . . 168
8.9 Примеры последовательностей сообщений запрос-ответ . . . 189
Приложение A (справочное) Сводная таблица требований . . . 197
Приложение B (обязательное) Профили сертификатов . . . 203
Приложение C (справочное) Применение сертификатов . . . 213
Приложение D (справочное) Шифрование для распределения секретных ключей . . . 224
Приложение E (справочное) Обзор XML-подписей . . . 226
Приложение F (рекомендуемое) Определения схем . . . 230
Приложение G (обязательное) Требования к идентификаторам . . . 258
Приложение H (справочное) Задание последовательности сообщений для пересогласования . . . 260
Приложение I (справочное) Привязка имен элементов сообщений ISO 15118 к терминам SAE J2847/2 . . . 264
Приложение J (справочное) Примеры сообщений . . . 268
Приложение K (справочное) Привязка элементов случаев использования части 1 . . . 291
Приложение ДА (справочное) Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте . . . 331
Приложение ДБ (справочное) Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта . . . 332
Библиография . . . 333

Введение

Международная организация по стандартизации (ИСО) объединяет национальные организации по стандартизации (комитет — член ИСО). Разработка международных стандартов осуществляется, как правило, техническими комитетами ИСО. Каждый комитет — член ИСО, заинтересованный в участии в проектах по тематике, для которой был создан технический комитет, имеет право быть представленным в этом комитете. Международные правительственные и неправительственные организации, связанные с ИСО, также принимают участие в работе объединения. ИСО непосредственно сотрудничает с Международной электротехнической комиссией (МЭК) по всем вопросам стандартизации электротехнических изделий.

Методики, использованные для разработки настоящего стандарта и для его дальнейшего применения, содержатся в директивах ИСО/МЭК, часть 1. В частности, следует отметить различные критерии, используемые для утверждения разных типов документов ИСО. Настоящий стандарт составлен в соответствии с редакторскими правилами части 2 директив ИСО/МЭК (gost.e-ecolog.ru).

Следует иметь в виду, что некоторые положения настоящего стандарта могут быть объектом патентных прав. ИСО не несет ответственности за идентификацию какого-либо одного или всех патентных прав. Детали любого патентного права, выявленного при разработке стандарта, должны находиться в введении и/или в перечне полученных патентных заявок ИСО (gost.e-ecolog.ru

Любое оригинальное наименование, используемое в настоящем стандарте, является информацией для удобства пользователей.

Толкование значений специфических терминов ИСО и выражений, относящихся к оценке соответствия, а также информация о строгом соблюдении ИСО принципов Всемирной торговой организации (ВТО) в отношении технических барьеров в торговле (ТБТ) содержатся по URL-адресу: gost.e-ecolog.ru foreword.html.

Настоящий стандарт курирует Технический комитет ISO 22 «*Дорожные транспортные средства*», подкомитет SC 3 «*Электрическое и электронное оборудование*».

Настоящий стандарт разработан при сотрудничестве с Техническим комитетом МЭК/ТК 69 «Электромобили и грузовые электрокары промышленного назначения».

ИСО 15118 под общим названием «Транспорт дорожный. Интерфейс связи автомобиль — электрическая сеть» состоит из следующих частей:

- часть 1: Общая информация и определение случаев использования;
- часть 2: Требования к протоколу сетевого и прикладного уровней;
- часть 3: Требования к физическому и канальному уровням.

Продолжающийся энергетический кризис, а также необходимость снижения выброса загрязняющих веществ, вызывающих парниковый эффект, заставили производителей предпринять серьезные усилия по снижению энергопотребления выпускаемых транспортных средств. В настоящее время производители заняты разработкой транспортных средств, передвигающихся частично или полностью на электрической тяге. Такие транспортные средства уменьшат зависимость транспорта от ископаемых источников энергии, улучшат глобальную энергоэффективность экономики и сократят выброс в атмосферу двуокиси углерода от дорожного транспорта в случае, если электроэнергия будет получена из возобновляемых источников. Для заряда аккумуляторных батарей, установленных на борту подобных транспортных средств, потребуются специальные инфраструктурные объекты.

Большая часть работы по стандартизации размеров и спецификации электрического оснащения инфраструктурных объектов зарядных станций, а также интерфейсов транспортных средств уже реализована в соответствующих стандартах или группах МЭК. Однако вопрос, касающийся передачи информации от транспортного средства на местный объект и в сеть, пока не проработан на должном уровне.

Связь «автомобиль — электрическая сеть» дает преимущества для оптимизации энергоресурсов сети и системы выработки электроэнергии. Это объясняется возможностью подзаряжать транспортные средства в оптимальный, с точки зрения энергосбережения, промежуток времени. Следует разработать эффективную и удобную систему оплаты. Необходимый канал связи может использоваться в будущем для стабилизации электрической сети, а также для предоставления дополнительных информационных услуг, необходимых для эффективной эксплуатации электромобилей.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ТРАНСПОРТ ДОРОЖНЫЙ

Интерфейс связи автомобиль — электрическая сеть

Часть 2

Требования к протоколу сетевого и прикладного уровней

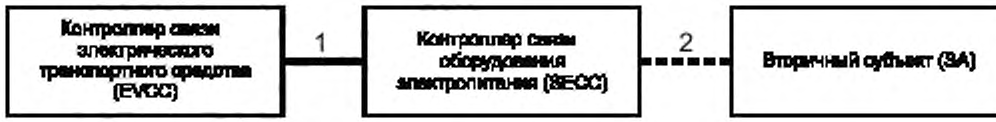
Road vehicles. Vehicle-to-Grid Communication Interface. Part 2. Network and application protocol requirements

Дата введения — 2019—06—01

1 Область применения

Настоящий стандарт определяет связь между аккумуляторными электромобилями (BEV), гибридными автомобилями с подзарядкой от электросети (PHEV) и оборудованием электроснабжения электромобиля (EVSE). Набор команд для приложения, требования к которому устанавливает настоящий стандарт, служит для поддержки передачи энергии от EVSE к электроавтомобилю (EV). *ГОСТ Р 58122 (ИСО 15118-1:2013)* содержит дополнительные элементы случаев использования (часть 1, идентификаторы элементов случаев использования: F4 и F5), описывающие двунаправленную передачу энергии. Реализация указанных случаев использования требует оптимизации набора команд для приложения, описываемого в настоящем стандарте.

Настоящий стандарт распространяется на сети между EV (BEV или PHEV) и EVSE и устанавливает требования к особенностям обнаружения EV в коммуникационной сети и подключения по интернет-протоколу (IP) между EVCC и SECC *(см. рисунок 1)*.



1 — предмет настоящего стандарта;
2 — определение сообщений учитывает случаи использования, которые определены для связи между SECC и SA

Рисунок 1 — Коммуникационные отношения между EVCC, SECC и вторичным субъектом

В настоящем стандарте описаны сообщения, модель данных, формат представления данных на базе XML/EXI, использующих протоколы связи V2GTP, безопасности транспортного уровня TLS, управления передачей TCP и IPv6. Также в нем описан доступ к сервисам канального уровня с точки зрения уровня 3. Функциональность канального уровня и физического уровня описана в *[1]*.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 58122—2018 (ИСО 15118-1:2013) Транспорт дорожный. Интерфейс связи автомобиль — электрическая сеть. Часть 1. Общая информация и определение случаев использования

ГОСТ Р МЭК 61851-1—2013 Система токопроводящей зарядки электромобилей. Часть 1. Общие требования

ГОСТ Р МЭК 62196-1—2013 Вилки, штепсельные розетки, соединители и вводы для транспортных средств. Кондуктивная зарядка для электромобилей. Часть 1. Общие требования

ГОСТ Р МЭК 62196-2—2013 Вилки, штепсельные розетки, соединители и вводы для транспортных средств. Кондуктивная зарядка для электромобилей. Часть 2. Требования размерной совместимости и взаимозаменяемости для штыревых разъемов и арматуры сети переменного тока

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены термины по *ГОСТ 58122 (ИСО 15118-1)*, а также следующие термины с соответствующими определениями:

3.1 **базовая зарядка;** BC: Этап зарядки во время сеанса зарядки, управляемый в соответствии с *ГОСТ Р МЭК 61851-1*.

3.2 **предельные параметры зарядки:** Набор физических ограничений (например, напряжение, сила тока, энергия, мощность), которые согласуются во время сеанса связи транспортного средства и электросети для сеанса зарядки.

3.3 **таймер установления связи:** Таймер, отслеживающий время от подключения до сообщения об установлении сеанса.

3.4 **сертификат контракта:** Сертификат, выдаваемый EVCC либо корневым удостоверяющим органом (CA) коммуникации транспортного средства и электросети или подчиненным удостоверяющим органом (Sub-CA), который используется в XML-подписях на уровне приложения для того, чтобы SECC или вторичный субъект мог проверить контракт, выданный EVCC, и подписи, выданные EVCC.

3.5 **состояние линии управления:** Состояние EV, передаваемое по линии управления в соответствии с *ГОСТ Р МЭК 61851-1*.

3.6 **удостоверение:** Что-либо, обеспечивающее основу для уверенности, доверия, кредита и т. д.

Пример — Сертификаты, пароли, имена пользователя и т. д.

3.7 **установление канала связи:** Этап установления канала обмена данными.

Примечание 1 — Условие входа: любой действительный сигнал линии управления в соответствии с ГОСТ *Р МЭК 61851-1*; условие выхода: D-LINK_READY.indication(DLINKSTATUS=LinkEstablished).

3.8 **особые правила кодирования = правило кодирования ASN-1;** DER: Метод кодирования объекта данных, например сертификата X.509, подлежащего подписанию цифровой подписью или проверке подписи.

3.9 **глобальный адрес:** IP-адрес с неограниченным охватом.

3.10 **зарядка со связью по протоколу высокого уровня;** HLC-C: Один из этапов зарядки в соответствии *с настоящим стандартом*.

3.11 **локальный адрес канала:** IP-адрес, действующий только для данной линии связи, который может быть использован для доступа к смежным интерфейсам, подключенным к той же линии связи.

3.12 **идентификационный режим:** Обязательные и факультативные сообщения и параметры для идентификации, относящиеся к сценариям зарядки, которые используют внешние средства идентификации (EIM), и к сценариям зарядки, использующим режим «Plug and Charge» (PnC).

Примечание 1 — Идентификационный режим охватывает набор схожих сценариев зарядки для конкретного средства идентификации.

3.13 **IP-адрес:** Идентификатор межсетевого уровня для интерфейса или набора интерфейсов.

3.14 **максимальная единица передачи;** MTU: Максимальный размер (в байтах) наибольшего блока протокольных данных, который канальный уровень может передать.

3.15 **набор сообщений:** Набор обязательных сообщений при взаимодействии транспортного средства и электросети (V2G) и параметров для EVCC или SECC, охватывающий один или несколько элементов случаев использования.

3.16 **таймер сообщений:** Таймер, осуществляющий мониторинг пары запрос-ответ.

3.17 **сетевой сегмент:** Совокупность устройств, которые могут обмениваться данными на канальном уровне непосредственно через канальные адреса.

Пример — Локальная сеть Ethernet: все устройства, которые могут видеть друг друга посредством MAC-адресов.

3.18 **узел:** Устройство, которое реализует IPv6.

3.19 **сертификат изготовителя:** Сертификат, выдаваемый EVCC для того, чтобы сертификат контракта мог быть надежно запрошен и получен от вторичного субъекта.

3.20 **время исполнения:** Нефункциональное временное требование, определяющее время, которое не должно быть превышено объектом V2G при исполнении или обработке определенного задания.

Примечание 1 — Это фиксированное значение времени.

3.21 **частная среда:** Зона с ограниченным (физическим) доступом для небольшого числа транспортных средств (EV). Данная зона может быть частным парковочным гаражом или гаражом/парковочной стоянкой компании, владеющей собственным парком EV. Вместо общественных зарядных станций EVSE могут использоваться один или несколько частных настенных зарядных блоков. Для того чтобы частный настенный зарядный блок был простым и дешевым в производстве и эксплуатации, ему разрешается постоянно оставаться в режиме офлайн. Это позволяет частному настенному зарядному блоку использовать листовые сертификаты с более длительным максимальным сроком действия, чем разрешаемый для публичных зарядных станций, а также использовать личный корневой сертификат, который отличается от корневых сертификатов V2G и который должен быть установлен на каждое EV. Если зарядка EV разрешена в данной конкретной частной зоне, то результатом будет ограничение числа EV, принадлежащих к отдельной частной среде. При этом отличие от «надежной среды» заключается в том, что в частной (собственно частной, а не обладающей дополнительной «надежностью») среде TLS, где соответствующее шифрование данных на уровне соединения всегда используется, обработка сертификатов упрощена для частного настенного зарядного модуля (EVSE), поскольку он может находиться в режиме офлайн постоянно, обеспечивая неограниченные сроки действия сертификатов, более короткую длину цепочки сертификатов, исключение OCSP и дополнительного «режима сопряжения».

3.22 **идентификационный режим:** Группа обязательных и факультативных наборов сообщений, охватывающая набор схожих сценариев зарядки для конкретного средства идентификации.

3.23 **пересогласование:** Обмен сообщениями для актуализации графика зарядки, согласованного между EV и EVSE во время сеанса связи V2G, путем повторной передачи параметров SASchedule и ChargingProfile.

3.24 **пара сообщений запрос-ответ:** Сообщение-запрос и соответствующее сообщение-ответ.

3.25 **последовательность сообщений запрос-ответ:** Предопределенная последовательность пар сообщений запрос-ответ.

3.26 **клиент SDP:** Субъект V2G, который использует сервер SDP, чтобы получить информацию о конфигурации SECC для обеспечения возможности доступа к SECC.

3.27 **сервер SDP:** Субъект V2G, предоставляющий информацию о конфигурации для доступа к SECC.

3.28 **сертификат SECC:** Сертификат, выдаваемый SECC либо корневым CA V2G, либо Sub-CA, который используется в TLS для того, чтобы EVCC мог проверить аутентичность SECC.

3.29 **таймер последовательности:** Таймер, осуществляющий мониторинг последовательности сообщений запрос-ответ.

3.30 **Sub-CA:** Подчиненный удостоверяющий орган, который выдает сертификаты SECC и/или сертификаты контракта от имени корневого CA V2G.

Примечание 1 — Полномочие на выдачу сертификатов делегируется корневым CA V2G, и он может отозвать сетрификаты Sub-CA в любое время.

3.31 **сертификат Sub-CA:** Сертификат, выдаваемый органом Sub-CA.

3.32 **TCP_DATA:** Порт/интерфейс для передачи данных на базе TCP-соединения.

3.33 **тайм-аут:** Требование, определяющее время, в течение которого объект V2G осуществляет мониторинг системы связи до наступления определенного события.

Примечание 1 — Если заданное время превышено, соответствующий объект V2G инициирует обработку связанной с этим ошибки. Это фиксированное значение времени.

3.34 **таймер:** Устройство или элемент программного обеспечения, используемый для измерения времени.

Примечание 1 — В зависимости от конкретного случая использования таймер применяется для запуска определенных системных элементов.

3.35 **безопасная зона:** Закрытая пользовательская группа (например, участники системы по совместному пользованию транспортными средствами) с некоторым предварительно распределенным средством доступа к зарядному сервису SECC (например, ключ от домашнего гаража, радиочастотный брелок для совместного пользования транспортными средствами), т. е. группа, за которую кто-то отвечает, например владелец гаража, оператор системы совместного пользования транспортными средствами или оператор такси.

3.36 **цикл V2G:** Этап обмена сообщениями V2G для управления процессом зарядки в соответствии с настоящим стандартом.

3.37 **сеанс связи V2G:** Ассоциация двух конкретных субъектов V2G для обмена сообщениями V2G.

3.38 **субъект V2G:** Первичный субъект, участвующий в связи V2G с помощью обязательного или дополнительного коммуникационного протокола, требования к которому установлены в настоящем стандарте

3.39 **сообщение V2G:** Сообщение, обмен которым происходит на уровне приложения.

Примечание 1 — См. раздел 8 «Сообщения прикладного уровня».

3.40 **установление V2G:** Этап установления обмена сообщениями V2G.

Примечание 1 — Условие входа: D-LINK_READY.indication(DLINKSTATUS=LinkEstablished); условие выхода: PowerDeliveryReq с ChargeProgress равен Start или Stop.

3.41 **коммуникационный протокол V2G:** Коммуникационный протокол для передачи сообщений V2G между двумя субъектами V2GTP.

3.42 **субъект V2GTP:** Субъект V2G, поддерживающий коммуникационный протокол V2G.

3.43 **корневой CA V2G:** Удостоверяющий орган (CA), который выдает сертификаты контракта и/или сертификаты SECC либо делегирует полномочие на выдачу таких сертификатов Sub-CA.

3.44 **корневой сертификат V2G:** Сертификат, выдаваемый корневому CA V2G.

4 Обозначения и сокращения

В настоящем стандарте применены следующие обозначения и сокращения:

BEV — аккумуляторное электрическое транспортное средство;

CA — удостоверяющий орган;

CRL — перечень аннулированных сертификатов;

DH — криптографический протокол Диффи — Хеллмана;

DER — особые правила кодирования;

ECDSA — алгоритм цифровой подписи на основе эллиптических кривых;

EMAID — идентификатор аккаунта EVEV;

EMOCH — центр обработки данных оператора услуг для EV [см. также *ГОСТ Р 58122 (15118-1:2013)*];

EV — электрическое транспортное средство,

EVCC — контроллер связи электрического транспортного средства;

EVSE — оборудование электропитания электрических транспортных средств;

EXI — эффективный XML-обмен;

OCSP — протокол статуса онлайнового сертификата;

OEM — изготовитель транспортных средств;

NACK — отрицательное квитирование;

PDU — блок протокольных данных;

PHEV — подзаряжаемое гибридное электрическое транспортное средство;

PKI — инфраструктура открытых ключей;

PLC — связь по силовой линии электропередачи;
PnC — «Plug and Charge» («Подключай и заряжай»);
SA — вторичный субъект;
SDP — протокол обнаружения SECC;
SDU — блок сервисных данных;
SECC — контроллер связи оборудования электропитания;
TLC — безопасность транспортного уровня;
TCP — протокол управления передачей;
V2G — взаимодействие транспортных средств и электросети;
V2G CI — интерфейс связи взаимодействия транспортных средств и электросети;
V2GTP — коммуникационный протокол V2G;
UDP — протокол пользовательских дейтаграмм;
XML — eXtensible Markup Language (расширяемый язык разметки).

5 Понятия

5.1 Определение сервисов на базе стандарта взаимодействия открытых систем (OSI)

Настоящий стандарт базируется на понятиях сервисов OSI (см. *[2]*) применительно к отдельным уровням, описанным в настоящем стандарте.

Настоящий стандарт описывает требования, применяющиеся к уровням 3—7 в соответствии с уровневой архитектурой OSI.

5.2 Структура требований

В настоящем стандарте используется определенная структура требований, а именно каждое индивидуальное требование, включенное в настоящий стандарт, идентифицируется уникальным номером. Данная структура требований облегчает отслеживание требований и установление требований к тестовым сценариям. Применяется следующий формат:

«[V2G"Y"-"XXX"]» — текст требования, где:

- «V2G» *характеризует объект требований настоящего стандарта*,
- Y идентифицирует *номер части настоящего стандарта*,
- XXX представляет собой номер индивидуального требования и
- «текст требования» включает фактический текст требования.

Пример — [V2G2-000] — это пример требования.

5.3 Использование RFC-ссылок

При использовании RFC-ссылок все требования «должен/не должен» являются обязательными.

Примечание — Номера требований описаны в приложении А.

[V2G2-001] В настоящем стандарте, если RFC-ссылка была обновлена одним или несколькими RFC, то изменение применяется полностью.

[V2G2-002] Если изменение или часть изменения, применяющаяся к RFC-ссылке, не совместима с исходным RFC или реализацией, описываемой настоящим стандартом, то обновление не действует.

[V2G2-003] Все опубликованные списки ошибок для RFC-ссылок полностью действительны для настоящего стандарта.

5.4 Представление, используемое для диаграмм XML-схем

В настоящем стандарте в качестве формата для описания сообщений V2G используется XML. Подробности относительно обозначений XML-схем, используемых в настоящем стандарте, приведены в руководстве *[3]*.

Для облегчения различения типов определений, используемых в XML-схемах, в настоящем стандарте для имен приняты следующие правила:

- у комплексных типов первые буквы прописные;
- у простых типов первые буквы строчные.

6 Обзор документов

На рисунке 2 показаны требования, содержащиеся в настоящем стандарте, *ГОСТ Р 58122 (ИСО 15118-1:2013) и [1]*, и их распределение в соответствии с уровневой архитектурой OSI.

Полужирными рамками выделены требования, применяющиеся к уровням 3—7 в соответствии с уровневой архитектурой OSI, светлыми рамками — требования уровней 1 и 2, включая стандартизованный интерфейс сервисных примитивов V2G, данные в *[1]*.

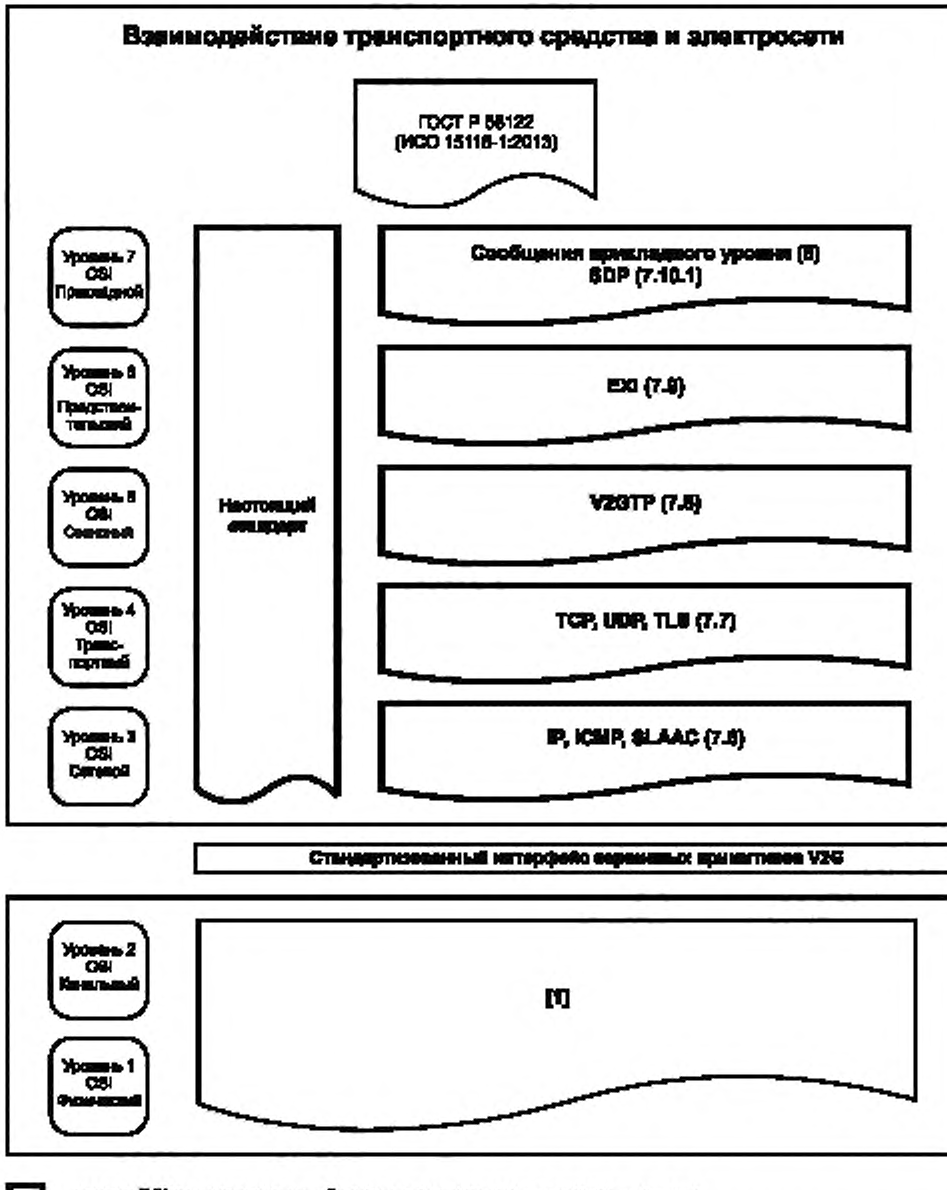


Рисунок 2 — Обзор связи документов V2G

7 Базовые требования к связи V2G

7.1 Общая информация

В настоящем стандарте описана реализация элементов случаев использования V2G, определения которых даны в *ГОСТ Р 58122 (ИСО 15118-1:2013)*.

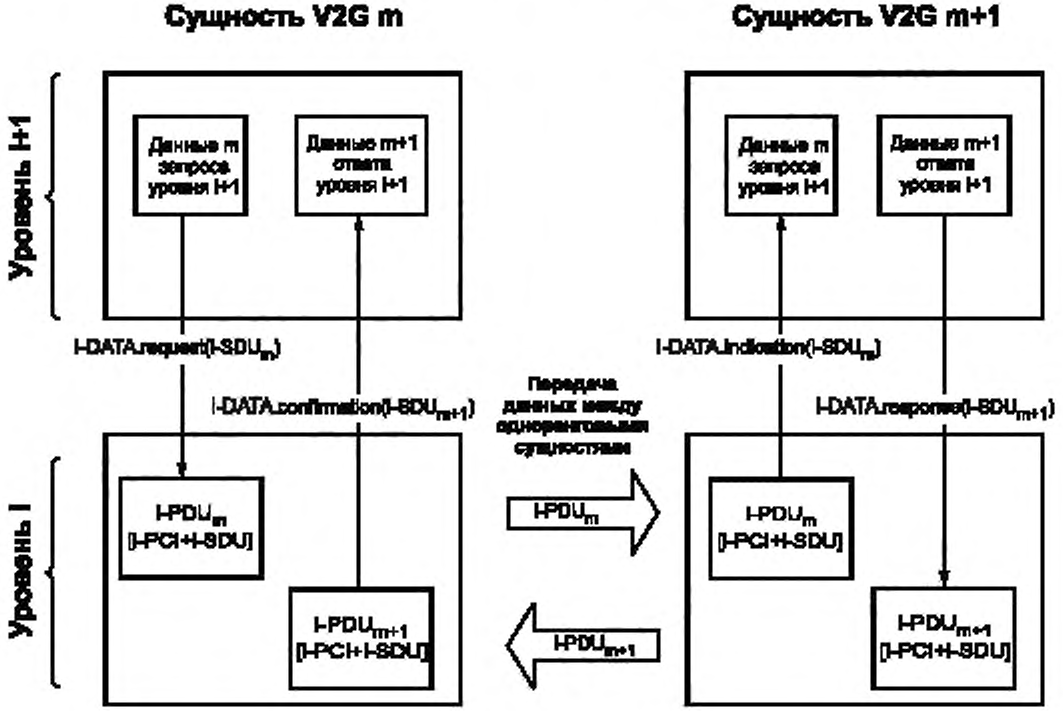
7.2 Концепция сервисного примитива уровневой архитектуры OSI

7.2.1 Обзор

В данном пункте описана уровневая архитектура OSI, применяемая в настоящем стандарте. Она призвана служить простым средством описания интерфейсов между отдельными уровнями протокола связи, требуемыми настоящим стандартом, а также позволяет более точно определить требования синхронизации.

Требования к сервисам устанавливаются путем описания сервисных примитивов и параметров, характеризующих сервис. Это абстрактное определение сервисов, не требующее конкретной реализации.

На рисунке 3 показан упрощенный вид взаимодействия уровней OSI, достаточный для понимания принципов уровневой архитектуры OSI в контексте настоящего стандарта.



PDU — блок протокольных данных сетевого объекта;
PCI — информация управления протоколом;
SDU — блок сервисных данных сетевого объекта

Рисунок 3 — Принципы уровневой архитектуры OSI

Когда экземпляр m субъекта V2G уровня i+1 обменивается данными с экземпляром m+1 субъекта V2G уровня i+1, каждый экземпляр использует сервисы экземпляра уровня i. Сервис определяется как набор сервисных примитивов.

7.2.2 Синтаксис сервисных примитивов

Сервисные примитивы описываются следующим синтаксисом.

[Инициал уровня]-[ИМЯ].[тип примитива](список параметров), где

- [инициал уровня] один из следующих семи: [Физический, Канальный, Сетевой, Транспортный, Сеансный, Представительский, Приложение];
- [ИМЯ] есть имя примитива.

Пример — Типичными примерами для поля [ИМЯ] являются СОЕДИНИТЬ, РАЗЪЕДИНИТЬ, ДАННЫЕ. В настоящем стандарте и в [1] используются также другие имена;

- [тип примитива] один из следующих четырех: [запрос, индикация, ответ, подтверждение];
- (список параметров) включает список отделяемых запятой параметров, которые пользователь сервиса должен предоставлять при использовании соответствующего сервисного примитива; факультативные параметры отмечены скобками «[..]».

Примечание — В настоящем стандарте тип примитива «indication» (индикация) всегда указывает на событие асинхронно по отношению к верхнему уровню.

7.3 Концепция безопасности

7.3.1 Потоки вызовов (блок-схемы)

На рисунках 4 и 5 проиллюстрированы примеры связи в полуонлайновом и онлайновом режимах с точки зрения безопасности и показаны необходимые применяемые сервисы безопасности, а также абстрактный вид различных данных, необходимых для работы.

Полные схемы потока данных/последовательности приведены в подразделе 8.8 настоящего стандарта. На данных обзорных рисунках следует выделять только информацию, относящуюся к безопасности.

Концепция безопасности обеспечивает базовый защитный механизм для осуществления связи в сети таким образом, чтобы предотвратить несанкционированный доступ. Для определенных сценариев требуется применение безопасности транспортного уровня (TLS) для связи между EVCC и SECC. Для некоторых других сценариев использование TLS является факультативным. Конкретные сообщения защищаются на уровне приложения (XML-сообщения), если данные необходимо защитить на пути от вторичного субъекта или к нему или если защита должна существовать дольше, чем существует TLS. Кроме того, концепция не зависит от дополнительных защитных механизмов на уровнях ниже уровня 3 уровневой модели OSI.

На рисунке 4 показан пример использования дежурного соединения для режима PnC.

В режиме идентификации PnC вся связь на базе TCP/IP защищается с помощью аутентифицируемого в одностороннем порядке канала TLS между двумя одноранговыми субъектами (TLS не обязательна для некоторых режимов, кроме режима идентификации PnC).

Конечным пунктом всей связи является SECC. Показания прибора учета периодически подписываются транспортным средством для поддержки процесса расчета (см. 8.4.3.13.1). Эта информация может быть использована для расчета, если местные правила это разрешают. EVSE предоставляет учетные данные о зарядке, содержащие подписанные показания прибора учета для дальнейшей обработки.

Примечание 1 — Связь между SECC и SA на рисунке 4 показана только с целью информации и не служит для установления конкретной последовательности сообщений.

На рисунке 5 показан пример в онлайновой режиме связи для случая использования PnC.

Как и в полуонлайновом режиме, в данном примере в режиме PnC вся связь на базе TCP/IP защищается с помощью аутентифицируемого в одностороннем порядке канала TLS между двумя одноранговыми сущностями (TLS не обязательна для некоторых режимов, кроме режима PnC).

Может потребоваться передача вторичному субъекту для дальнейшей обработки части информации, предоставляемой транспортным средством, например удостоверений транспортного средства, чтобы обеспечить возможность подписания информации о тарифе. EVCC рассчитывает профиль зарядки (см. 8.4.3.9.2) и посылает его SECC, который может направить его системам SA. Дальнейший процесс аналогичен полуонлайновому режиму, за исключением того, что окончательные данные о зарядке могут быть предоставлены SA напрямую. Предполагается, что SECC будут также использовать безопасное транспортное соединение с SA, хотя безопасность связи транспортного средства с SA обеспечивается на прикладном уровне.

Примечание 2 — Связь между SECC и SA на рисунке 5 показана только с целью информации и не служит для установления конкретной последовательности сообщений.

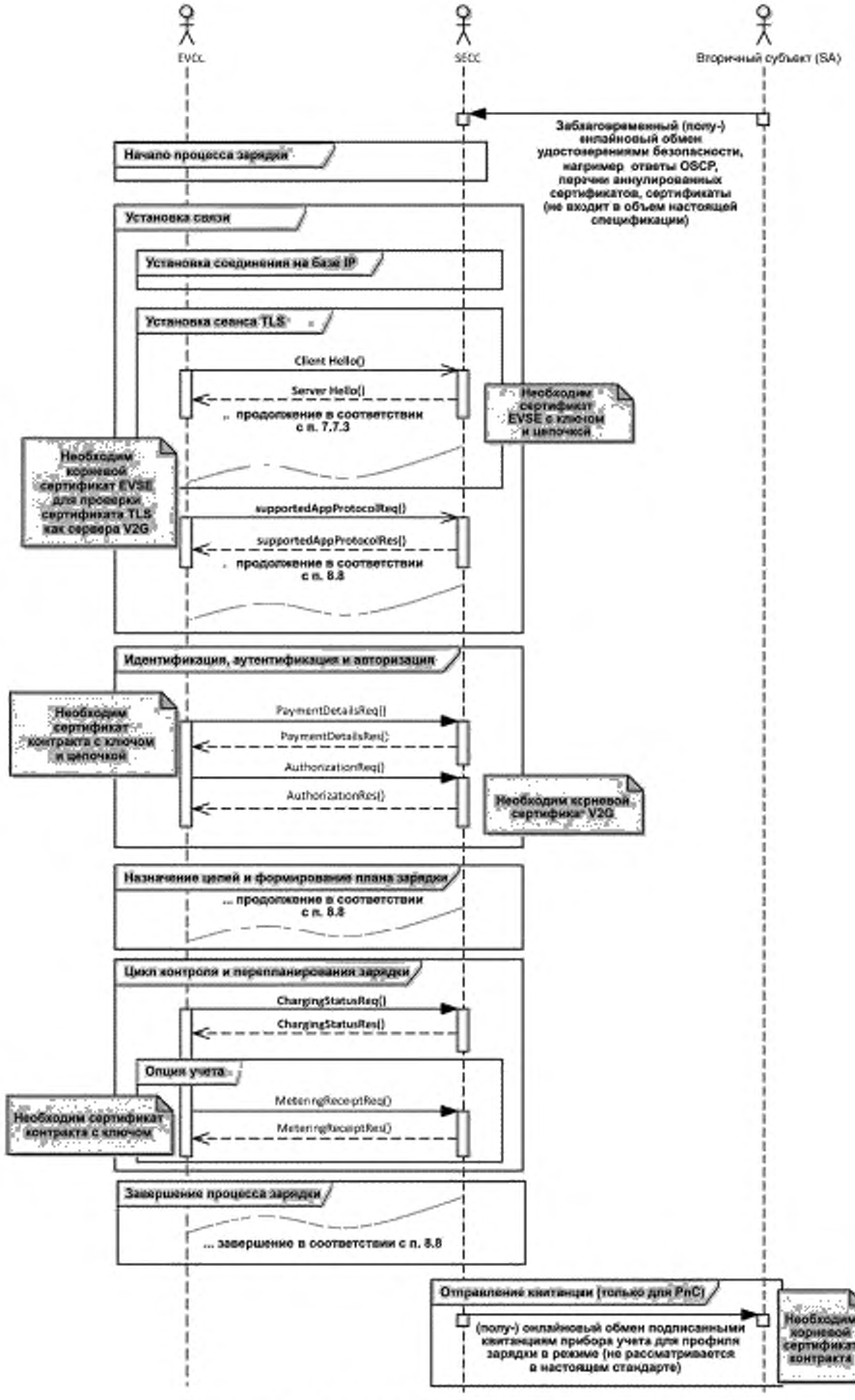


Рисунок 4 — Пример связи в полуонлайновом режиме

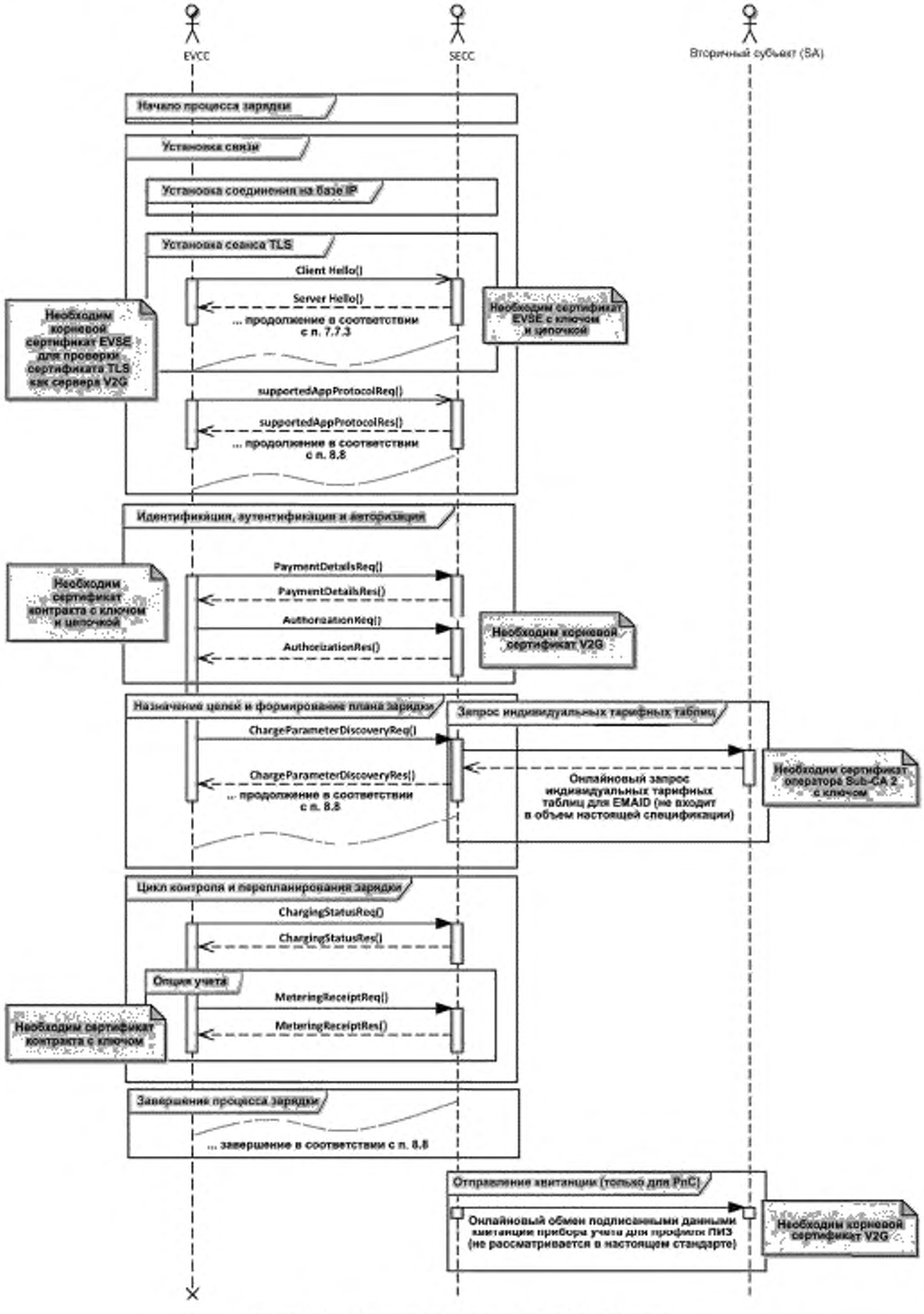


Рисунок 5 — Пример связи в онлайновом режиме

Имеются случаи дополнительного использования связей, в которых механизмы безопасности необходимы на уровне приложения. Речь идет о начальной регистрации контрактных ключей и сертификата, а также об обновлении сертификата. В этих случаях используются ключи изготовителя для конкретного транспортного средства. Обзор всех необходимых сертификатов содержится в приложении В.

7.3.2 Управление сертификатами и ключами

Потоки вызовов, описанные в 7.3.1, требуют наличия нескольких сертификатов, применяемых для разных уровней безопасности. Используют сертификаты SECC, применяемые на уровне TLS для их аутентификации EVCC, сертификаты контрактов, применяемые на уровне приложения для аутентификации SECC и/или вторичного субъекта, корневые сертификаты V2G и сертификаты Sub-CA, которые удостоверяют сертификаты SECC и сертификаты контрактов.

Кроме вышеуказанных сертификатов, могут использоваться корневые сертификаты изготовителя и сервисные сертификаты изготовителя, применяемые для установки и обновления сертификатов контрактов.

Обзор всех возможных профилей приведен в приложении В.

[V2G2-004] Каждый субъект V2G должен использовать сертификаты X.509v3 вследствие необходимости расширений для хранения параметров криптосистемы на основе эллиптических кривых. Дополнительные сведения см. в *[4]*.

В таблице 1 показано, из каких полей состоит сертификат X.509v3.

Таблица 1 — Базовые поля сертификата

Поле сертификата	Описание
Версия	Версия сертификата (для настоящего стандарта должна быть v3 = 0x2)
Серийный номер	Уникальный номер сертификата (в домене эмитента)
Алгоритм подписи	Используемый алгоритм подписи
Эмитент	Субъект, который выпустил и подписал сертификат
Срок действия	Период времени, в течение которого сертификат действителен
Субъект	Субъект, которому выдан сертификат
Открытый ключ	Открытый ключ, соответствующий закрытому ключу
Уникальный идентификатор эмитента	Факультативный уникальный идентификатор эмитента
Уникальный идентификатор субъекта	Факультативный уникальный идентификатор субъекта
Расширения	Дополнительно (см. таблицу 2)
Подпись	Подпись сертификата, генерируемая эмитентом

Примечание 1 — Информацию о полях сертификата см. в *[4]*.

В зависимости от случая использования сертификата X.509v3 в него может быть включена дополнительная информация о так называемых расширениях сертификатов. В таблице 2 обобщены распространенные расширения сертификатов.

Таблица 2 — Примеры расширений сертификатов

Расширения сертификатов	Описание
Применение ключей	Использование соответствующего закрытого ключа (например, цифровая подпись, невозможность отказа, шифрование ключей)
Расширенное применение ключей	Дальнейшее описание применения ключей с использованием идентификаторов объекта, например: аутентификация сервера (1.3.6.1.5.5.7.3.1), аутентификация клиента (1.3.6.1.5.5.7.3.2). Примечание — Иногда также кодируются следующие значения: Netscape SGC (1.3.6.1.4.1.311.10.3.3),

Окончание таблицы 2

Расширения сертификатов	Описание
Расширенное применение ключей	Microsoft SGC (2.16.840.1.113730.4.1). SGC расшифровывается как Server Gated Crypto (управляемая сервером криптография) и показывает, что сервер может также использовать криптографию для связи с браузером клиента. Это расширение использовалось во времена строгого криптографического экспортного контроля для обеспечения надлежащей защиты передачи данных финансовым веб-сайтом
Точка рассылки CRL	Место получения списков отозванных сертификатов
OCSP	Место получения ответа OCSP (существует только для сертификатов Sub-CA). См. подробности в *[5]*
Авторизационная информация	Дополнительная авторизационная информация
Альтернативное имя субъекта	Альтернативные имена субъекта
Базовое ограничение = CA	При условии, что сертификат является корневым сертификатом V2G или сертификатом Sub-CA

Примечание 2 — Информация об идентификаторах объектов, например 1.3.6.1.5.5.7.3.1, приведена в *[6]*.

[V2G2-005] Каждый субъект должен поддерживать хэш-функцию SHA-256 (процесс подписания) в соответствии с *[7]* [для идентификатора в случае использования по *ГОСТ Р 58122 (ИСО 15118-1:2013)*: F1].

[V2G2-006] Для каждого субъекта V2G процесс подписания должен быть на базе криптографии с использованием эллиптических кривых (secp256r1[представление SECG]) с алгоритмом подписи ECDSA [для идентификатора в случае использования по *ГОСТ Р 58122 (ИСО 15118-1:2013)*: F1].

[V2G2-007] Длина ключа для асимметричной криптографии на основе эллиптических кривых, используемой каждой сущностью V2G, должна быть 256 бит.

[V2G2-885] Все валидации сертификатов должны осуществляться в соответствии с *[4]*. EVCC и SECC могут помещать в кэш результаты валидации сертификатов во время одного сеанса зарядки.

[V2G2-926] Должны поддерживаться расширения сертификатов, указанные в приложении B. Отклонения указаны, где это необходимо.

[V2G2-925] Листовой сертификат должен рассматриваться как недействительный, если якорь доверия в конце цепочки не соответствует конкретному корневому сертификату, необходимому для определенного использования, или если необходимое значение компонента домена отсутствует.

[V2G2-910] EVCC должен реализовывать механизм проверки сроков действия сертификатов и ответов OCSP.

[V2G2-886] EVCC может выбирать точность своего источника времени по своему усмотрению, но он должен соблюдать выбранную точность при использовании этого источника времени. Точность должна быть по крайней мере равна одному дню.

Примечание 3 — Это требование теоретически даже позволяет применять точность, например до одного года. Реализующий должен тщательно взвесить последствия в отношении безопасности такого решения.

Примечание 4 — Если соединение TLS успешно установлено, EVCC может использовать время от поля EVSETimeStamp, посылаемое от SECC через указанное соединение, для синхронизации своего внутреннего генератора синхроимпульсов, однако реализующий должен тщательно взвесить последствия в отношении безопасности такого решения.

[V2G2-887] SECC должен обеспечивать наличие для себя в любой момент времени сертификата со сроком действия не менее чем на один час больше.

7.3.3 Число корневых сертификатов и срок их действия, глубина и размер сертификата

[V2G2-008] Каждый EVCC должен поддерживать не менее одного корневого сертификата V2G.

[V2G2-877] Каждый SECC должен поддерживать хранение не менее чем 10 корневых сертификатов V2G.

Примечание 1 — Ввиду наложения сроков действия могут существовать до 10 одновременно действительных сертификатов (см. **[V2G2-012]**).

Примечание 2 — Настоятельно рекомендуется число корневых сертификатов V2G, равное 5. При этом только один является обязательным. Наличие меньше пяти или только одного сертификата связано с риском для изготовителя. Наличие только одного корневого сертификата V2G позволяет осуществлять зарядку транспортного средства только в одном регионе. Изготовитель должен будет указать в руководстве и во время предложения транспортного средства потребителям, что зарядка транспортного средства возможна только в «домашнем» регионе. Если изготовитель опасается, что пяти корневых сертификатов V2G недостаточно для покрытия «радиуса использования» его транспортных средств, он может обеспечить больше мест хранения корневых сертификатов.

Примечание 3 — Предполагается, что корневые сертификаты V2G, применяемые на уровне 4 OSI, также используются на уровне 7 OSI. Предполагается также, что в Европе будет один удостоверяющий орган для корневых сертификатов, аналогичный доверительному центру, который был создан для EURO5/EU5, что другие регионы мира будут также иметь удостоверяющий орган для корневых сертификатов, охватывающих большее число субъектов. Это позволяет предположить, что пяти корневых сертификатов для пяти регионов мира достаточно. Если будет принято решение о дополнительном пространстве для сертификатов, то это не повлияет на совместимость. Для SECC, однако, является обязательным хранение большего числа сертификатов, поскольку может существовать 10 одновременно действующих корневых сертификатов для каждого региона мира.

[V2G2-009] Ограничение длины пути дерева сертификата PKI должно составлять 3.

Примечание 4 — Ограничение длины пути определяет число несамоподписных сертификатов в пути удостоверения, т. е. имеется до трех уровней сертификатов, производных от корневого сертификата.

[V2G2-010] Размер сертификата в кодированной по DER форме должен быть не больше 800 байтов. Для передачи все сертификаты должны быть кодированными по DER.

[V2G2-011] Срок действия любых корневых сертификатов V2G должен быть 40 лет.

[V2G2-012] В любой момент времени должен быть в наличии корневой сертификат V2G, действительный для каждого CA корневых сертификатов V2G по крайней мере в течение последующих 35 лет.

Примечание 5 — Пояснения относительно срока действия сертификата, числа корневых сертификатов, глубины и размера сертификата приведены в С.1 (приложение С).

[V2G2-878] Максимальное число одновременно действительных корневых сертификатов V2G для одного корневого CA никогда не должно быть больше 10.

[V2G2-867] Корень V2G не должен выдавать листовой сертификат.

Примечание 6 — Только Sub-CA могут выдавать листовые сертификаты.

[V2G2-869] CA или Sub-CA не должен выпускать и подписывать сертификат, который действителен в течение срока действия, который больше, чем срок действия его самого как подписывающего субъекта.

[V2G2-911] Если используется только один уровень Sub-CA, т. е. Sub-CA, подписанный корневым CA, непосредственно подписывает листовые сертификаты, профиль Sub-CA 2 должен действовать для данного Sub-CA.

Следующие требования обеспечивают упрощенную обработку сертификатов в условиях частного пользования. Более подробное пояснение приведено в С.2 (приложение С).

[V2G2-882] Каждый EVCC должен быть способен хранить не менее одного корневого сертификата частного оператора, который должен быть заменяем владельцем транспортного средства.

[V2G2-927] EVCC должен рассматривать хранимые корневые сертификаты частного оператора как якоря доверия для проверки сертификатов SECC.

[V2G2-883] Сроки действия сертификатов SECC, имеющих надежный корневой сертификат частного оператора в качестве их якоря доверия, могут быть выбраны подписывающим сертификат на его усмотрение.

[V2G2-868] Для частной среды органы Sub-CA и ответы OCSP не поддерживаются.

Примечание 7 — Это означает, что корневой сертификат частного оператора напрямую подписывает листовой сертификат.

7.3.4 Поддержка и применение TLS

[V2G2-630] Поддержка TLS обязательна для EVCC для всех идентификационных режимов, кроме идентификационного режима EIM с набором сообщений зарядки переменным током с EIM и набором сообщений зарядки постоянным током с EIM, исключая в обоих случаях набор сообщений дополнительных услуг (ДУ), как указано в 8.6.

[V2G2-631] Поддержка TLS обязательна для SECC для всех идентификационных режимов, кроме идентификационного режима EIM в безопасной среде с набором сообщений зарядки переменным током с EIM и набором сообщений зарядки постоянным током с EIM, исключая в обоих случаях набор сообщений ДУ, как указано в 8.6.

На базе установления TCP или TLS на транспортном уровне действуют следующие ограничения:

[V2G2-632] Если используется сеанс связи без TLS, SECC должен только обеспечивать идентификационный режим EIM и наборы сообщений зарядки переменным током с EIM или зарядки постоянным током с EIM, указав «ExternalPayment» в параметре PaymentOptionList сообщения ServiceDiscoveryRes, как указано в 8.4.3.3.3.

[V2G2-633] Если используется сеанс связи без TLS, EVCC должен принимать только идентификационный режим EIM с наборами сообщений зарядки переменным током с EIM или зарядки постоянным током с EIM независимо от предложения SECC также других идентификационных режимов и наборов сообщений.

[V2G2-634] Если используется сеанс связи без TLS, SECC не должен выдавать наборы сообщений PnC.

[V2G2-635] Если используется сеанс связи без TLS, EVCC не должен применять наборы сообщений PnC.

[V2G2-636] Если используется сеанс связи без TLS, SECC не должен выдавать набор сообщений ДУ.

[V2G2-637] Если используется сеанс связи без TLS, EVCC не должен применять набор сообщений ДУ.

[V2G2-638] Меры безопасности для дополнительных услуг, разрешаемых набором сообщений ДУ (предлагаемых в ServiceDiscoveryRes), не рассматриваются в настоящем стандарте.

[V2G2-639] Если TLS не применяется, обе стороны должны обеспечить невозможность изменения в текущем сеансе зарядки идентификационного режима EIM на другой идентификационный режим и изменения набора сообщений зарядки переменным током с EIM или набора сообщений постоянным током с EIM на другой набор сообщений (чтобы избежать снижения безопасности сеанса в результате изменений).

[V2G2-640] Если применяется TLS, обе стороны должны обеспечить, чтобы выключение TLS приводило к завершению сеанса зарядки (чтобы избежать снижения безопасности сеанса в результате изменений).

Поддержка, использование и ограничения TLS приведены в таблицах 3 и 4.

Таблица 3 — Поддержка TLS для идентификационных режимов EIM

	Другая среда (небезопасная)				Безопасная среда[c]			
Разрешенные наборы сообщений[a]	Постоянный ток с EIM	Переменный ток с EIM	Постоянный ток с EIM, ДУ	Переменный ток с EIM, ДУ	Постоянный ток с EIM	Переменный ток с EIM	Постоянный ток с EIM, ДУ	Переменный ток с EIM, ДУ
EVCC Поддержка TLS	Факультативно	Факультативно	Обязательно	Обязательно	Факультативно	Факультативно	Обязательно	Обязательно
SECC Поддержка TLS	Обязательно	Обязательно	Обязательно	Обязательно	Факультативно	Факультативно	Обязательно	Обязательно
Согласование использования TLS с помощью SDP[b]	EV решает, EVSE должно принять решение EV	EV решает, EVSE должно принять решение EV	EV должно использовать TLS, EVSE должно отклонить, если EV не использует TLS	EV должно использовать TLS, EVSE должно отклонить, если EV не использует TLS	См. таблицу 4	См. таблицу 4	EV должно использовать TLS, EVSE должно отклонить, если EV не использует TLS	EV должно использовать TLS, EVSE должно отклонить, если EV не использует TLS

Окончание таблицы 3

[a] Относится к сообщениям, описанным в настоящем стандарте. В случае ДУ, конечно, возможны другие соединения (которые, не входят в настоящий стандарт).
[b] Отклонение означает прекращение коммуникации.
[c] См. определение в *[1]*.

Примечание 1 — Отсутствие поддержки TLS SECC может привести в общем случае к прекращению сеанса зарядки конкретных EV, поскольку за одобрение сеансов без TLS отвечает EV.

Таблица 4 — Подтверждение SDP при зарядке переменным и постоянным током с идентификационным режимом EIM в безопасной среде

Поддержка TLS EVCC	Поддержка TLS SECC	Запрос SDP EVCC	Ответ SDP от SECC
EVCC имеет поддержку TLS	SECC имеет поддержку TLS	EVCC дает сигнал TLS	SECC должен ответить TLS
		EVCC дает сигнал TCP	SECC должен ответить TCP
	SECC не имеет поддержки TLS	EVCC дает сигнал TLS	SECC может показать, что он не поддерживает TLS, показав «0x10 = нет безопасности транспортного уровня» в его ответе SDP. EV может принять решение о своем желании установить TCP без TLS или прекратить установление связи
		EVCC дает сигнал TCP	SECC должен ответить TCP
EVCC не имеет поддержки TLS	SECC имеет поддержку TLS	EVCC дает сигнал TCP	SECC должен ответить TCP
	SECC не имеет поддержки TLS	EVCC дает сигнал TCP	SECC должен ответить TCP

Примечание 2 — Для набора сообщений «EIM AC» («зарядка переменным током с EIM») и «EIM DC» («зарядка постоянным током с EIM») EV отвечает за решение о неиспользовании TLS и принятие риска небезопасного соединения.

Примечание 3 — Если применяется TLS, то она всегда используется с односторонней аутентификацией (со стороны EVSE). Если TLS не применяется, аутентификация EV со стороны EVSE, а также дополнительная защита канала связи на транспортном уровне отсутствуют.

Примечание 4 — Меры в отношении каких-либо связанных с функциональной безопасностью рисков вследствие перенапряжений и перегрузок по току (случайных или преднамеренных) должны приниматься путем реализации соответствующих стандартов электробезопасности (например, *ГОСТ Р МЭК 61851-1, [8], [9],[10]*).

Меры в отношении каких-либо связанных с функциональной безопасностью рисков вследствие перенапряжений и перегрузок по току должны приниматься в соответствии с перечисленными ниже требованиями:

[V2G2-643] EVSE должно поддерживать меры безопасности, описанные в *ГОСТ Р МЭК 61851-1* и *[8]*, для защиты от перенапряжений и перегрузок по току при использовании идентификационного режима EIM для зарядки переменным током.

[V2G2-644] EVSE должно поддерживать меры безопасности, описанные в *ГОСТ Р МЭК 61851-1* и *[9]*, для защиты от перенапряжений и перегрузок по току при использовании идентификационного режима EIM для зарядки постоянным током.

7.4 Состояния связи V2G и работа с каналом данных

Обмен сообщениями V2G на уровне приложения требует установления протоколов нижнего уровня для обеспечения обмена сообщениями V2G между EVCC и SECC. В настоящем стандарте рассматриваются упомянутые выше протоколы канального уровня. Канальный уровень описан в *[1]*.

В данном подразделе описываются базовые требования для установления связи на уровне выше канального уровня. Настоящий стандарт не описывает какую-либо их реализацию. На рисунках данного подраздела показана только базовая последовательность для управления протокольным пакетом.

Значения таймеров, тайм-аута и времени исполнения, используемые в данном подразделе, описаны в 8.8. На рисунке 6 показан обзор коммуникационных состояний EVCC. К EVCC применяются следующие требования:

[V2G2-014] После установления соединения на канальном уровне (D-LINK_READY. indication(DLINKSTATUS=Link established) EVCC должен инициировать механизм присваивания IP-адресов, как указано в 7.6.3.2 и 7.6.3.3.

[V2G2-016] EVCC должен отработать механизм присваивания IP-адресов, как указано в 7.6.3.2 и 7.6.3.3.

[V2G2-017] EVCC должен остановить механизм присваивания IP-адресов, как указано в 7.6.3.2 и 7.6.3.3, когда V2G_EVCC_CommunicationSetup_Timer равен или больше V2G_EVCC_ CommunicationSetup_Timeout.

[V2G2-018] После присвоения локального IP-адреса канала EVCC должен начать процесс обнаружения адреса SECC, как указано в 7.10.1.

[V2G2-019] EVCC должен запустить клиента SDP в соответствии с 7.10.1.

[V2G2-645] В зависимости от протокола безопасности и транспортного протокола, запрошенного в сообщении-запросе SDP, ожидаемого протокола безопасности и транспортного протокола, отправленного сервером SDP, EV должно принять решение о применении TLS.

[V2G2-020] EVCC должен остановить SDP, когда V2G_EVCC_CommunicationSetup_Timer равен или больше V2G_EVCC_CommunicationSetup_Timeout.

[V2G2-021] Если EV принимает решение о применении соединения с обеспечением безопасности, EVCC должен установить соединение TLS с SECC, как описано в 7.7.3, после обнаружения IP-адреса SECC, порта, имеющихся транспортного протокола и опций безопасности.

[V2G2-646] Если EV принимает решение о применении соединения без обеспечения безопасности, EVCC должен установить подключение TCP к SECC, как описано в 7.7.1, после обнаружения IP-адреса SECC, порта, имеющихся транспортного протокола и опций безопасности.

[V2G2-022] В зависимости от **[V2G2-021]** и **[V2G2-646]** EVCC должен попытаться установить соединение с TLS или TCP в соответствии с 7.7.3.

[V2G2-023] EVCC должен прекратить попытки установить соединение с TLS или TCP, когда V2G_EVCC_CommunicationSetup_Timer равен или больше V2G_EVCC_ CommunicationSetup_Timeout.

[V2G2-024] После установления соединения с TLS или TCP EVCC должен инициировать сеанс связи V2G, как описано в разделе 8.

[V2G2-025] EVCC должен прекратить соединение с TLS или TCP после остановки сеанса связи V2G.

[V2G2-717] Если EVCC отправил сообщение SessionStopReq с параметром ChargingSession, равным «Terminate», он должен прекратить Data-Link (D-LINK_TERMINATE.request()) после получения сообщения SessionStopRes.

[V2G2-718] Если EVCC отправил сообщение SessionStopReq с параметром ChargingSession, равным «Pause», он должен сделать паузу Data-Link (D-LINK_PAUSE.request()) после получения сообщения SessionStopRes и продолжить выполнение **[V2G2-014]**.

[V2G2-719] Каждый раз, когда EVCC получает указание на отсутствие канала обмена данными (D-LINK_READY.indication (DLINKSTATUS=No link), он должен продолжить выполнение **[V2G2-014]**.

[V2G2-720] Если EVCC определяет какую-либо ошибку, он должен указать на ошибку канала обмена данными (D-LINK_ERROR.request()) и продолжить выполнение **[V2G2-014]**.

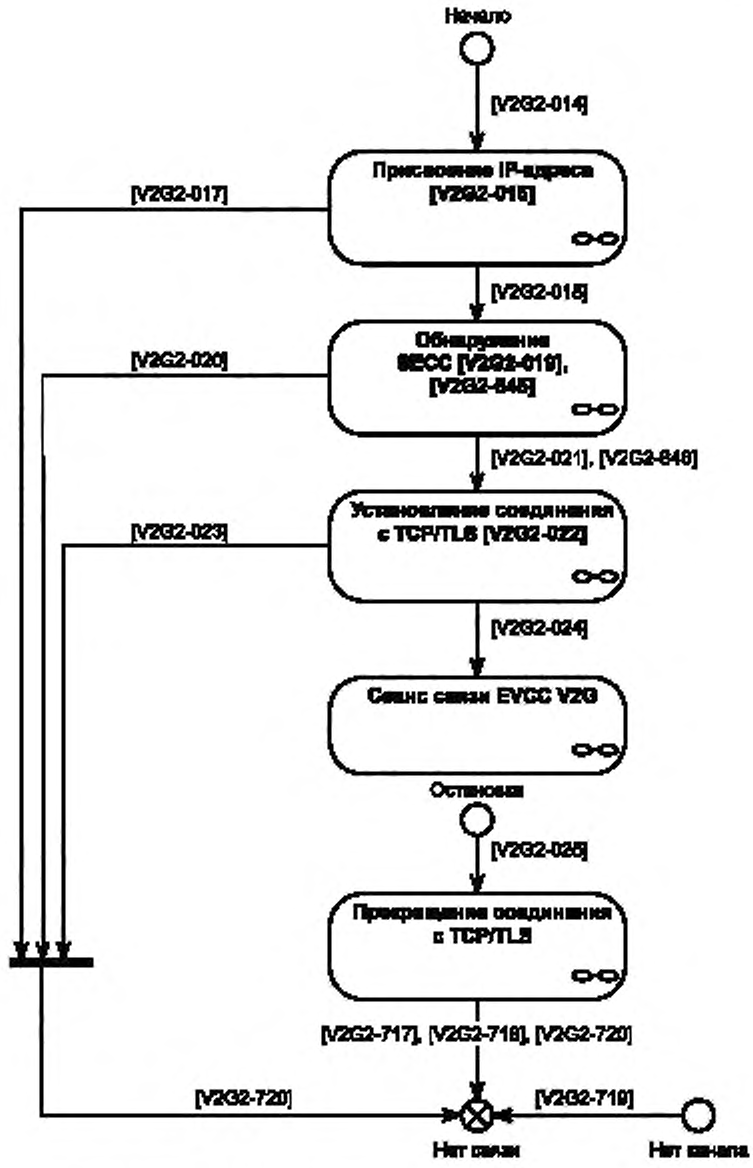


Рисунок 6 — Обзор коммуникационных состояний EVCC

На рисунке 7 показан обзор состояний SECC во время сеанса ккоммуникации V2G.

К SECC применяются следующие требования:

[V2G2-721] После указания на успешное установление канала обмена данными (D-LINK_READY. indication(DLINKSTATUS=Link established) SECC должен инициировать механизм присваивания адресов.

[V2G2-026] SECC должен сконфигурировать IP-адрес (статический или динамический) с помощью любого надлежащего механизма.

Примечание 1 — Описание присвоения адреса SECC для его интеграции в различные коммуникационные инфраструктуры не рассматривается в настоящем стандарте. Это позволяет оператору самому назначить надлежащий механизм. Например, оператор может выбрать любой существующий механизм, дающий действительный IP-адрес, как, например, статический IP, или механизм, описанный в 7.6.3.2 и 7.6.3.3.

Примечание 2 — SECC должен быть сконфигурирован с действительным IP-адресом для обеспечения соединения с EVCC. Механизм присвоения IP-адресов для SECC не влияет на совместимость. В настоящем стандарте он не рассматривается.

[V2G2-027] После присвоения IP-адреса SECC должен запустить сервер SDP, как указано в 7.10.1.

Примечание 3 — Не требуется, чтобы сервис обнаружения SECC реализовывался непосредственно в SECC. Возможно также наличие отдельного устройства, обеспечивающего сервис обнаружения SECC.

[V2G2-723] SECC должен остановить сервер SDP, когда SECC_CommunicationSetup_Timer равен или больше V2G_SECC _CommunicationSetup_Performance_Time.

[V2G2-029] SECC должен остановить механизм присваивания IP-адресов, когда V2G_SECC_ CommunicationSetup_Timer равен или больше V2G_SECC_CommunicationSetup_ Performance_Time.

[V2G2-030] После того как сервер SDP успешно запущен, SECC должен ожидать инициализации соединения с TLS или TCP в зависимости сообщения-ответа SDP, как указано в 7.10.1.5.

[V2G2-031] SECC должен ожидать установления соединения с TLS или TCP.

[V2G2-032] SECC должен прекратить ожидание установления соединения с TLS, когда V2G_SECC_ CommunicationSetup_Timer равен или больше V2G_SECC_CommunicationSetup_ Performance_Time.

[V2G2-033] После установления соединения с TLS или TCP SECC должен ожидать инициализации сеанса связи V2G, как описано в разделе 8.

[V2G2-722] Если **[V2G2-033]** применимо, SECC может остановить сервер SDP после успешного установления соединения с TLS или TCP.

[V2G2-034] SECC должен прекратить соединение с TLS или TCP после остановки сеанса связи V2G.

[V2G2-724] Если SECC получил сообщение SessionStopReq с параметром ChargingSession, равным «Terminate», он должен прекратить работу с каналом обмена данными (D-LINK_TERMINATE.request()) после отправления сообщения SessionStopRes.

[V2G2-725] Если SECC получил сообщение SessionStopReq с параметром ChargingSession, равным «Pause», он должен сделать паузу в работе с каналом обмена данными (D-LINK_PAUSE.request()) после отправления сообщения SessionStopRes и продолжить выполнение **[V2G2-721]**.

[V2G2-726] Каждый раз, когда SECC получает указание на отсутствие канала обмена данными (D-LINK_READY.indication (DLINKSTATUS=No link), он продолжает выполнение **[V2G2-721]**.

[V2G2-727] Если SECC определяет какую-либо ошибку, он должен указать на ошибку канала обмена данными (D-LINK_ERROR.request()) и продолжить выполнение **[V2G2-721]**.

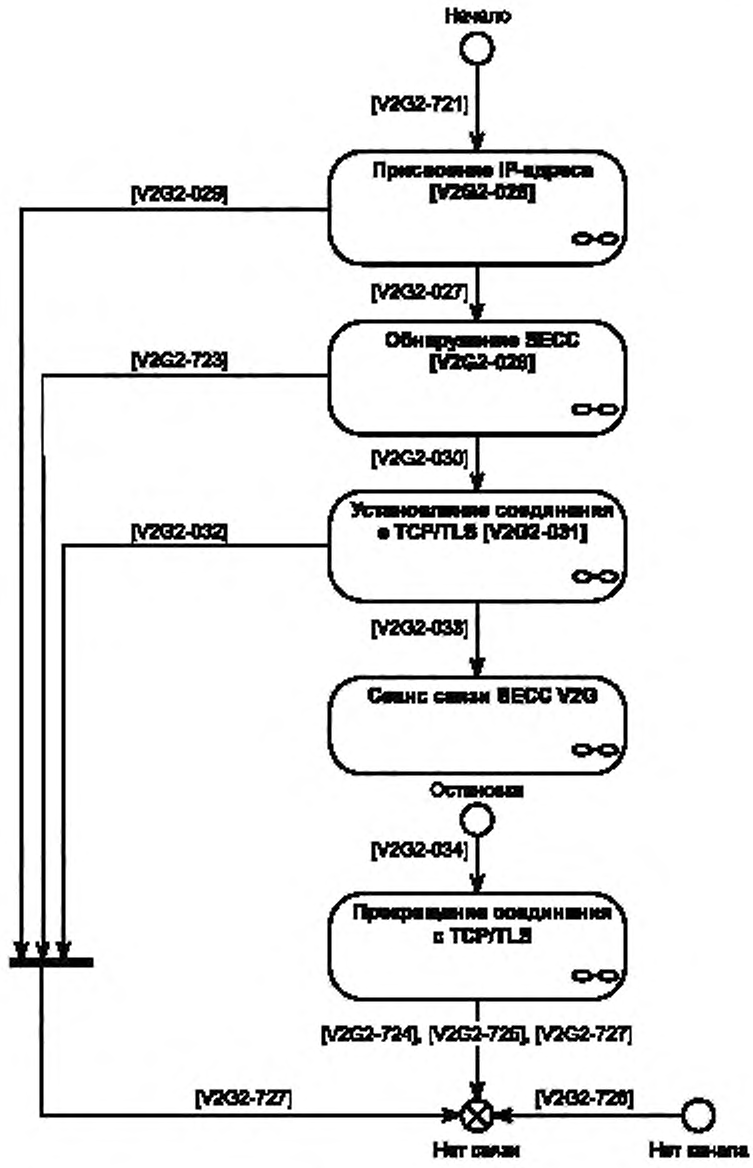


Рисунок 7 — Обзор состояний SECC во время сеанса коммуникации V2G

7.5 Канальный уровень

Канальный уровень поддерживает транспорт пакетов IP. В *[1]* описаны необходимые дополнительные сведения о канальном уровне.

[V2G2-035] Если EVCC осуществляет связь через PLC, EVCC должен соответствовать *[1]*.

[V2G2-036] Если SECC осуществляет связь через PLC, SECC должен соответствовать *[1]*.

7.6 Сетевой уровень

7.6.1 Общие положения

Протокол, описанный в настоящем стандарте, базируется на интернет-протоколе, известном как IPv6 (см. *[11]*).

7.6.2 Применимые нормы RFC, ограничения и настройки параметров протокола

7.6.2.1 IPv6

[V2G2-037] Субъект V2G должен поддерживать IPv6 в соответствии с *[11]*.

[V2G2-038] Хотя *[11]* описывает IPsec как обязательный, от субъекта V2G не требуется реализация IPsec.

[V2G2-039] Ни от какого субъекта V2G не требуется реализации заголовка маршрутизации RH0, как указано в *[12]*.

Примечание 1 — Указания по назначению для поля типа маршрутизации в заголовке маршрутизации IPv6 даны в *[13]*. Рекомендуется придерживаться этих указаний.

[V2G2-040] Субъект V2G должен реализовывать определение MTU в соответствии с *[14]*.

[V2G2-041] Субъект V2G должен поддерживать обработку наложений фрагментов IP в соответствии с *[15]*.

Примечание 2 — Субъект V2G должен соответствовать спецификации *[16]*, которая дополняет спецификацию в соответствии с *[11]*.

[V2G2-042] При отправлении пакета IPv6 от EVCC к SECC или от SECC к EVCC IP-фрагментация не должна использоваться.

Примечание 3 — Связь между EVCC и вторичными субъектами не рассматривается в настоящем стандарте. Она может использовать или не использовать IP-фрагментацию.

7.6.2.2 Протокол динамического управления хостами (DHCPV6)

Требования к каналу обмена данными описаны в *[1]*. EVCC начинает присвоение адресов, запускаемое каналом обмена данными, когда канальное соединение установлено. Это выполняется в соответствии с 7.6.3.2 с использованием SLAAC, которое является обязательным в соответствии с настоящим стандартом. DHCPv6 может быть реализован как факультативный метод конфигурирования IP.

[V2G2-043] Если EVCC решает ввести в действие клиента DHCPv6, то он должен осуществлять это, как описано в *[17]*.

[V2G2-044] Если инфраструктура, к которой подключено EV (EVSE), решает реализовать сервер DHCPv6, то она должна осуществлять это, как описано в *[17]*.

7.6.2.3 Обнаружение соседних объектов (OCO)

EVCC использует автоконфигурирование адресов без состояния для IPv6, чтобы генерировать адреса для его интерфейса. Все интерфейсы имеют локальный адрес канала. Для обеспечения уникальности адресов и для поддержки глобальных адресов используется протокол обнаружения соседних объектов.

[V2G2-045] EVCC должен реализовывать обнаружение соседних объектов в соответствии с *[18]*.

[V2G2-046] EVCC должен соответствовать *[19]*, допуская присвоение IP-адресов до завершения определения адресов-дубликатов.

7.6.2.4 Протокол управления сообщениями в сети Интернет (ICMP)

Протокол управления сообщениями в сети Интернет (ICMP) используется для отправки сообщений об ошибках (например, запрашиваемый сервис отсутствует, хост не доступен и т. д.).

[V2G2-047] Каждый субъект V2G должен реализовывать ICMPv6, как описано в *[20]*.

[V2G2-049] Каждый субъект V2G должен реализовывать нормы RFC, на которые даются ссылки в колонке «Ссылка» таблицы 5, описывающей детали реализации для соответствующих типов сообщений ICMP.

Таблица 5 — Обязательный набор сообщений протокола управления сообщениями в сети Интернет (ICMP)

Тип сообщения ICMP	Имя сообщения ICMP	Ссылка
1	Пункт назначения недостижим	*[20]*
2	Пакет слишком велик	*[20]*
3	Превышено время	*[20]*
4	Проблема параметра	*[20]*
128	Эхо запроса	*[20]*
129	Эхо ответа	*[20]*
133	Запрос маршрутизатора	*[18]*
134	Объявление маршрутизатора	*[18]*
135	Запрос соседнего объекта	*[18]*
136	Объявление соседнего объекта	*[18]*
137	Перенаправление сообщения	*[18]*
141	Обратное сообщение с запросом об обнаружении соседнего объекта	*[21]*
142	Обратное сообщение с объявлением об обнаружении соседнего объекта	*[21]*

7.6.3 IP-адресация

7.6.3.1 Общие положения

В настоящем пункте описывается, как EVCC получает действительные IP-адреса для связи по сети на базе IP. Следующие адреса учитываются в настоящем стандарте:

- локальный IP-адрес канала EVCC;
- глобальный IP-адрес EVCC, если маршрутизатор присутствует в локальном канале;
- IP-адрес SECC.

Примечание — Хост IPv6 может иметь несколько IP-адресов, присвоенных одному физическому интерфейсу сети, например локальный адрес канала и глобальный адрес.

7.6.3.2 Автоконфигурирование адресов без состояния (SLAAC)

[V2G2-050] Каждый субъект V2G должен поддерживать конфигурацию индивидуального локального адреса канала IPv6, как указано в *[22]*.

[V2G2-051] Идентификатор интерфейса локального адреса канала субъекта V2G должен генерироваться из его 48-битового MAC-идентификатора IEEE в соответствии с *[22]*.

[V2G2-052] EVCC должен поддерживать автоконфигурирование адресов IP6, как описано в *[23]*.

[V2G2-053] Если SECC выдает наборы сообщений установки сертификатов, обновления сертификатов или дополнительные услуги, как указано в 8.6.2, он должен поддерживать варианты объявления маршрутизатора IPv6 в соответствии с *[24]* для предоставления адреса сервера DNS.

7.6.3.3 Выбор адресов

[V2G2-054] Если поддерживаются несколько адресов IPv6, выбор адресов IPv6 по умолчанию должен выполняться в соответствии с *[25]*.

7.7 Транспортный уровень

7.7.1 Протокол управления передачей (TCP)

7.7.1.1 Обзор

Протокол управления передачей (TCP) позволяет приложениям субъектов V2G устанавливать надежное соединение для обмена данными с другими субъектами. Для надежного обмена и передачи данных от отправителя к получателю TCP дополнительно обеспечивает управление потоками и управление в условиях перегрузки, а также обеспечивает различные алгоритмы для осуществления контроля над перегрузкой и влиянием потоков.

7.7.1.2 Применимые RFC, ограничения и настройки параметров протокола (TCP)

[V2G2-055] Каждый субъект V2G должен осуществлять реализацию TCP, как указано в *[26]*.

7.7.1.3 Требования к производительности TCP и контрольной сумме

Следующие требования определяют подробности реализации TCP в отношении управления в условиях перегрузки, повторной передачи, хронирования, исходного размера окна и селективного подтверждения в целях улучшения эффективности работы TCP.

Рекомендуется использовать следующие алгоритмы повторной передачи дополнительно к стандартным методам TCP и управления в условиях перегрузки:

[V2G2-057] Каждый субъект V2G должен реализовывать управление TCP в условиях перегрузки в соответствии с *[27]*.

[V2G2-058] Каждый субъект V2G должен реализовывать модификацию NewReno к алгоритму быстрого восстановления TCP в соответствии с *[28]*.

[V2G2-059] Каждый субъект V2G должен рассчитать для TCP таймер повторной передачи в соответствии с *[29]*.

[V2G2-060] Для увеличения производительности TCP каждый субъект V2G должен реализовывать расширения TCP для повышения производительности в соответствии с *[30]*.

[V2G2-061] Каждый субъект V2G должен поддерживать опции избирательного подтверждения TCP в соответствии с *[31]*.

[V2G2-062] Каждый субъект V2G должен реализовывать опцию пользовательского тайм-аута в соответствии с *[32]*.

[V2G2-063] Указатель срочности для TCP не должен использоваться каким-либо субъектом V2G.

Рекомендуется использовать следующий алгоритм контрольной суммы:

[V2G2-064] Поля контрольной суммы, требующиеся в заголовках TCP, должны реализовываться в соответствии с *[33]*.

7.7.2 Протокол пользовательских дейтаграмм (UDP)

7.7.2.1 Обзор

Протокол пользовательских дейтаграмм (UDP) является протоколом без установления соединения. UDP не обеспечивает надежность и гарантии упорядочения, которые обеспечивает TCP. Пакеты могут приходить неупорядоченно или могут быть потеряны без уведомления отправителя или получателя. Однако UDP быстрее и более эффективен для многих легких или чувствительных ко времени задач. UDP находится на транспортном уровне модели уровневой архитектуры OSI.

Случаи использования UDP с механизмами безопасности описаны в *ГОСТ Р 58122 (ИСО 15118-1:2013)* (приложение В).

7.7.2.2 Применимые RFC, ограничения и настройки параметров протокола

[V2G2-065] Каждый субъект V2G должен осуществлять реализацию протокола пользовательских дейтаграмм в соответствии с *[34]*.

7.7.3 Безопасность транспортного уровня (TLS)

7.7.3.1 Обзор

Безопасность транспортного уровня обеспечивается путем применения TLS. Это позволяет установить аутентифицированный и криптографически защищенный (обеспечивающий защиту целостности и конфиденциальности) канал между EVCC и SECC. TLS обеспечивает одностороннюю или взаимную аутентификацию. Для данного подраздела настоящего стандарта было согласовано использование односторонней аутентификации (EVCC аутентифицирует SECC).

7.7.3.2 Применимые RFC

[V2G2-067] Односторонняя аутентификация с TLS версии 1.2 в соответствии с *[35]* с расширениями в соответствии с *[36]* должна поддерживаться каждым субъектом V2G. EVCC аутентифицирует SECC путем проверки цепочки сертификатов SECC, передаваемой от SECC к EVCC.

EVCC аутентифицирует SECC в соответствии с **[V2G2-067]**, что может быть также использовано для защиты циклического считывания показаний прибора учета между SECC и EVCC. Применение TLS с односторонней аутентификацией исключает необходимость дополнительной цифровой подписи показаний прибора со стороны SECC благодаря осуществлению безопасного сеанса.

Примечание 1 — В случае оплаты на SECC циклическое считывание показаний прибора учета на SECC прекращается и их не требуется посылать третьей стороне. Таким образом, отсутствует необходимость в последующей подписи данных, относящихся к расчету.

Применение односторонней аутентификации запрещает SECC проверять корректность EVCC. Поэтому SECC не может определить, является ли связывающийся EVCC аутентичным. Проверка корректности EVCC может быть реализована на уровне приложения, как описано в 7.9.2.

7.7.3.3 Применение безопасности транспортного уровня

[V2G2-068] SECC должен всегда действовать как компонент сервера TLS.

[V2G2-070] Каждый EVCC должен проверять действительность сертификатов Sub-CA (в цепочке сертификатов SECC) через ответ OCSP в соответствии с *[5]*, когда используется TLS. Запросы OCSP должны транспортироваться как часть подтверждения TLS с использованием расширения, указанного в *[37]*.

Примечание 1 — Не предусматривается, что EVCC запрашивает ответ OCSP для листового сертификата SECC во время подтверждения TLS.

«Режим сопряжения» поддерживает простое управление сертификатами в частной среде [см. С.2 (приложение С)].

[V2G2-870] «Режим сопряжения» EV может быть установлен с помощью фирменного механизма изготовителя. Данный «режим сопряжения» должен активироваться только определенным взаимодействием пользователя. Данный режим должен автоматически сбрасываться через 120 с.

Примечание 2 — «Режим сопряжения» может быть повторно активирован в любое время.

Примечание 3 — Изготовителю предлагается пояснить использование и риски «режима сопряжения», например, в руководстве пользователя.

[V2G2-649] SECC должен обновлять (помещать в кэш) ответ OCSP не реже чем раз в неделю. Одним из решений для обновления может быть, например, онлайновое соединение.

[V2G2-876] Срок действия ответа OCSP для Sub-CA 1 и Sub-CA 2 должен быть не дольше четырех недель.

Примечание 4 — Это увязывается со сроком действия листового сертификата SECC.

[V2G2-650] Хотя срок действия на EVCC не является обязательным, EVCC должен реализовывать механизм исключения устаревших сертификатов от SECC.

Примечание 5 — Обработка ошибок не рассматривается в настоящем стандарте. Обработка таких ошибок осуществляется по усмотрению изготовителя.

[V2G2-651] EVCC должен отправить список корневых сертификатов V2G, которые он имеет, расширение типа «trusted_ca_keys» в (расширенном) приветствии клиента в соответствии с *[36]*.

[V2G2-871] Если SECC развернут в среде, которая не является частной средой, он должен отправить свой собственный сертификат и цепочку до корня (исключая сам корневой сертификат), который должен быть одним из тех корней, на которые ранее указал EVCC как на присутствующие в EVCC. Если EVCC запросил ответы OCSP для каждого сертификата, которому сервер TLS отвечает, ответ OCSP должен быть отправлен EVCC в соответствии с *[5]* (пункт 4.2.1). SECC должен также направить сертификат отвечающего OCSP, если он не использует расширение в соответствии с *[37]*. Ответчик OCSP должен заполнить поле «certs» в структуре ответа OCSP (BasicOCSPResponse) соответствующим Sub-CA.

Примечание 6 — Ответчиком OCSP может быть сам Sub-CA или субъект, который непосредственно подписывается соответствующим Sub-CA с помощью пары ключей со специальным флагом расширенного использования ключа в сертификате.

[V2G2-923] Если SECC не может предоставить сертификат с цепочкой до одного из корней, на который указывал EVCC как на присутствующий в EVCC, SECC должен представить в составе подтверждения TLS действительный сертификат, включая цепочку до корневого сертификата (сам корневой сертификат не должен передаваться).

[V2G2-924] Если SECC выдал цепочку сертификатов, которая не прослеживается до корневого сертификата в списке надежных сертификатов, EVCC должен принять такой сертификат, только если он выполнил успешную валидацию цепочки сертификатов с помо-

щью механизма внешней валидации (например, протокол валидации сертификатов на базе сервера, простой протокол валидации сертификатов SCVP). Если валидация с помощью такого сервиса не выполняется, дает отрицательный результат или не удается (например, вследствие отсутствия соединения), EVCC должен рассматривать цепочку сертификатов как не прошедшую валидацию.

[V2G2-872] Если SECC развернут в частной среде, то он не должен отправлять ответ OCSP, а должен включить свой собственный частный корневой сертификат в цепочку сертификатов, которую он посылает.

Примечание 7 — Это поддерживает «режим сопряжения».

[V2G2-873] Если EVCC запросил ответ OCSP через «client hello» TLS и сервер TLS не отправил ответы OCSP в «server hello», EVCC должен выполнить следующее:
- если цепочка сертификатов сервера TLS содержит только один листовой сертификат и один корневой сертификат, который является одним из корневых сертификатов, который EV сохранил как частный корень, EVCC игнорирует тот факт, что сервер не отправил ответы OCSP;
- если цепочка сертификатов сервера TLS привязана к корню V2G, EVCC должен прервать соединение TLS.

[V2G2-874] Если режим сопряжения активирован, EVCC получает новый корневой сертификат от SECC во время подтверждения TLS и новый корневой сертификат выполняет требования для частного корневого сертификата (см. приложение B), то EVCC должен принять новый корневой сертификат, хранить его постоянно и обращаться с ним как с частным корневым сертификатом.

[V2G2-875] EVCC должен проверить сертификат SECC, полную цепочку сертификатов и все ответы OCSP (если применимо, включая действительность ответчика OCSP). EVCC должен также проверить, что сертификат SECC имеет компонент домена, установленный на «CPO». Если результат хотя бы одной из этих проверок будет отрицательным, EVCC должен прекратить процесс установления TLS и применить **[V2G2-023]**.

[V2G2-810] SECC должен поддерживать согласование максимальной длины фрагмента в соответствии с *[36]*.

[V2G2-811] SECC должен быть способен поддерживать максимальную длину фрагмента 2^9 байта, но не быть ограниченным этим числом, в соответствии с *[36]*.

7.7.3.4 Удостоверения транспортного уровня и наборы шифров безопасности

Для безопасности транспортного уровня EVCC аутентифицирует SECC с использованием сертификата SECC. Это достигается вследствие наличия у SECC закрытого ключа, который соответствует сертификату SECC, наличия у EVCC корневого сертификата V2G и проверки цепочки сертификатов от корневого сертификата V2G до сертификата SECC. Проверка действительности сертификата Sub-CA в цепочке сертификатов осуществляется через ответ OCSP, получаемый во время подтверждения TLS (подробности см. в *[36]*). Механизмы отзыва сертификатов SECC не требуются, но вместо этого требуются краткосрочные сертификаты.

SECC неизвестно, какой корневой сертификат V2G имеет EVCC из 10 действительных в настоящее время корневых сертификатов V2G для одного региона. Отсюда наличие 50 действительных в настоящее время корневых сертификатов V2G для всего мира. Поэтому необходимо, чтобы EVCC предоставил список корневых сертификатов V2G, которые он имеет, посредством подтверждения TLS. SECC выдает цепочку своего собственного сертификата SECC, корневой сертификат которого передается EVCC вместе с ответом OCSP для сертификатов Sub-CA в цепочке. Настоятельно рекомендуется обновлять ответ OCSP не реже чем раз в неделю.

Сообщение, передаваемое между EVCC и SECC, шифруется с помощью симметричного ключа (одностороннее согласование ключа), согласуемого во время этапа согласования ключа TLS. Поэтому сообщение не будет перехвачено, и EVCC и SECC могут быть уверены, что в течение сеанса партнер на самом деле тот же самый. (Перехвата сеанса не может быть.)

[V2G2-071] Каждый субъект V2G должен предоставлять удостоверения и информацию о безопасности, указанную в таблице 6, если используется TLS.

Таблица 6 — Аутентификация TLS

Аутентификация TLS	Требования к EVCC	Требования к SECC
Односторонняя (сторона сервера)	Наличие корневых сертификатов для проверки аутентичности сертификата SECC. Функциональная поддержка обработки запросов/ответов OCSP для проверки состояния валидации сертификата SECC во время подтверждения TLS	Ответ SECC в виде сертификата и соответствующего закрытого ключа OCSP для предоставления информации о состоянии валидации собственного сертификата SECC. SECC должен помещать в кэш и сохранять внутри ответы OCSP для собственной цепочки сертификатов Sub-CA регулярно (не реже чем еженедельно)

[V2G2-602] SECC должен поддерживать все наборы шифров, указанные в таблице 7, если используется TLS.

[V2G2-603] EVCC должен поддерживать не менее одного набора шифров, указанного в таблице 7, если используется TLS.

Дополнительные наборы шифров могут поддерживаться любым субъектом V2G.

Примечание — Изготовитель может принять решение о наборе шифра, поддерживаемом EVCC, на основе предлагаемых в таблице 7. Одного набора достаточно для компактных реализаций. Для функциональной совместимости SECC должен поддерживать все наборы шифров, указанные в таблице 7.

Таблица 7 — Поддерживаемые наборы шифров

Набор шифров	RFC
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	*[38]*
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	*[38]*

7.8 Коммуникационный протокол V2G

7.8.1 Общая информация

Коммуникационный протокол V2G (V2GTP) является компактным протоколом связи для передачи сообщений V2G между двумя субъектами V2GTP. Он в основном состоит из определения заголовка и полезных данных, что позволяет разделять и эффективно обрабатывать сообщения V2G. V2GTP является стандартным коммуникационным протоколом между EVCC и SECC, но может быть также использован для связи с другими субъектами V2G, которые поддерживают протокол V2GTP.

7.8.2 Поддерживаемые порты

V2GTP базируется на TLS+TCP, которые используют пару IP-адресов (адрес-источник и адрес назначения) и пару номеров портов (порт-источник и порт назначения) для установления и идентификации соединения. Соединение устанавливается от адреса-источника и порта-источника к адресу назначения и порту назначения. Порты, перечисленные в таблице 8, используются субъектами V2GTP.

Таблица 8 — Поддерживаемые порты TLS для V2GTP

Имя	Протокол	Номер порта	Описание
V2G_SRC_TCP_DATA	TCP (с адресацией конкретному устройству)	Номер порта в диапазоне динамических портов (49152-65535), как описано в *[39]*	Порт-источник V2GTP первичного субъекта (например, EVCC), который реализует протокол V2GTP
V2G_DST_TCP_DATA	TCP (с адресацией конкретному устройству)	Номер порта субъекта V2GTP, предоставляющего номер порта назначения в диапазоне динамических портов (49152-65535), как описано в *[39]*. Для SECC он присваивается динамически механизмом SDP (7.10.1)	Порт назначения V2GTP первичного субъекта (например, SECC)

Для субъектов V2GTP, реализующих V2GTP, действуют следующие общие требования:

[V2G2-073] Субъект V2GTP, предоставляющий порт назначения, должен поддерживать не менее одного соединения на локальном порте V2G_DST_TCP_DATA, как указано в таблице 8.

[V2G2-074] Субъект V2GTP, предоставляющий порт назначения, может поддерживать несколько одновременных соединений на локальном порте V2G_DST_TCP_DATA, как указано в таблице 8.

[V2G2-075] Субъект V2GTP, использующий порт-источник, должен поддерживать не менее одного соединения на локальном порте V2G_SRC_TCP_DATA, как указано в таблице 8.

[V2G2-076] Субъект V2GTP, использующий порт-источник, может поддерживать несколько соединений на локальном порте V2G_SRC_TCP_DATA, как указано в таблице 8.

Для EVCC и SECC особо действуют следующие требования:

[V2G2-077] EVCC должен использовать порт-источник V2G_SRC_TCP_DATA, как указано в таблице 8.

[V2G2-078] SECC должен предоставлять порт назначения V2G_DST_TCP_DATA, как указано в таблице 8.

[V2G2-079] EVCC должен поддерживать не менее одного соединения для сеанса связи V2G на порте V2G_SRC_TCP_DATA.

[V2G2-080] SECC должен поддерживать не менее одного соединения для сеанса связи V2G на порте V2G_DST_TCP_DATA.

[V2G2-081] EVCC должен использовать для соединения с SECC порт V2G_DST_TCP_DATA, выданный в последнем сообщении об обнаружении SECC (см. 7.10.1.5).

7.8.3 Блок протокольных данных

7.8.3.1 Структура PDU V2GTP

PDU V2GTP состоит из заголовка и основной части, или тела, как показано на рисунке 8.

Заголовок (8 байт)	Полезные данные (0-4294967295 байт)

Рисунок 8 — Структура сообщения V2GTP

Полезные данные содержат данные приложения (например, сообщение V2G). Заголовок отделяет полезные данные (т. е. индивидуальные сообщения V2GTP) в потоке байтов и дает информацию для обработки полезных данных.

Структура заголовка сообщения V2GTP показана на рисунке 9 и описана в таблице 9. Поддерживаемые типы полезных данных описаны в таблице 10.

№ байта	1	2	3	4	5	6	7	8
Поле заголовка	Версия протокола	Обратная версия протокола	Тип полезных данных	Длина полезных данных				

Рисунок 9 — Структура заголовка сообщения V2GTP

[V2G2-082] Субъект V2GTP должен использовать структуру заголовка, показанную на рисунке 9.

[V2G2-083] Субъект V2GTP должен отправить 8 байтов заголовка V2GTP в порядке, показанном на рисунке 9.

[V2G2-084] Байт с меньшим номером должен быть отправлен перед байтом с большим номером. Заголовок начинается с байта 1 и заканчивается байтом 8.

[V2G2-085] Субъект V2GTP должен отправлять поля «Тип полезных данных» и «Длина полезных данных» в формате с обратным порядком следования байтов: наиболее значимый байт передается первым, наименее значимый байт — последним.

Таблица 9 — Типовая структура заголовка V2GTP

Поле заголовка	Описание поля заголовка	Значения поля заголовка
Версия протокола	Идентифицирует версию протокола сообщений V2GTP	0x01: V2GTP версия 1 0x00, 0x02-0xFF: зарезервировано документом
Обратная версия протокола	Содержит обратное побитовое значение версии протокола и используется совместно с версией протокола V2GTP как схема проверки протокола, обеспечивающая получение правильно сформатированного сообщения V2GTP. Эквивалентно функции: <Protocol_Version> XOR 0xFF	0xFE: V2GTP версия 1
Тип полезных данных (GH_PT)	Содержит информацию о том, как декодировать полезные данные, следующие после заголовка V2GTP	Полный перечень значений типов полезных данных см. в таблице 10
Длина полезных данных (GH_PL)	Содержит длину полезных данных сообщения V2GTP в байтах (т. е. исключая байты типового заголовка V2GTP)	0…4294967295 (= <d>)

Таблица 10 — Обзор типов полезных данных V2GTP

Значение типа полезных данных	Имя типа полезных данных	Описывается в пункте
0x0000 – 0x8000	Зарезервировано	Неприменимо
0x8001	EXI-кодированное сообщение V2G	7.9.1.2
0x8002 – 0x8FFF	Зарезервировано	Неприменимо
0x9000	Сообщение с запросом SDP	7.10.1.4
0x9001	Сообщение с ответом SDP	7.10.1.5
0x9002 – 0x9FFF	Зарезервировано	Неприменимо
0xA000 – 0xFFFF	Специфичное использование изготовителя	Неприменимо

[V2G2-086] Субъект V2GTP должен использовать структуру сообщения V2GTP, показанную на рисунке 8, для отправления сообщения V2G, как описано в разделе 8.

[V2G2-087] Субъект V2GTP должен использовать определения, как указано в таблицах 9 и 10.

[V2G2-088] Для EXI-кодированных сообщений V2G (полезные данные 0x8001) субъект V2GTP должен использовать отдельное сообщение V2GTP для каждого сообщения V2G.

Примечание — Из требования **[V2G2-088]** вытекает, что поле полезных данных не может включать ни часть сообщения, ни несколько сообщений.

7.8.3.2 Обработка заголовка

Для обработки полезных данных субъект V2GTP должен сначала обработать заголовок. Для этого субъект V2GTP, получивший сообщение V2GTP, проверяет поле заголовка шаг за шагом. Описанная ниже обработка заголовка проиллюстрирована на рисунке 10.

[V2G2-089] Субъект V2GTP должен обработать заголовок V2GTP, как указано в таблице 9, до обработки полезных данных, как определено в таблице 10.

[V2G2-090] Субъект V2GTP должен проверить поля версии протокола и обратной версии протокола (последовательность синхронизирующих импульсов) перед какими-либо другими полями заголовка.

[V2G2-092] Субъект V2GTP должен проверить тип полезных данных после успешной проверки поля версии и обратной версии протокола.

[V2G2-094] Субъект V2GTP должен проверить длину полезных данных после успешной проверки типа полезных данных.

[V2G2-096] Если обработка заголовка была успешной, субъект V2GTP должен обрабатывать полезные данные.

[V2G2-800] Если заголовок V2GTP содержит неверные данные (например, не поддерживаемый тип полезных данных, неверную длину полезных данных или не поддерживаемую версию V2GTP), субъект V2GTP должен игнорировать это сообщение.

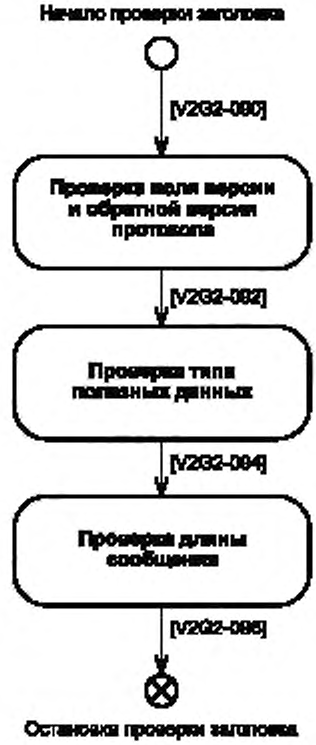


Рисунок 10 — Обработка типового заголовка V2GTP

7.9 Представительский уровень

7.9.1 XML и эффективный XML-обмен (EXI)

7.9.1.1 Обзор

Для описания набора сообщений V2G представительский уровень использует широко применяемое XML представление данных. В настоящем стандарте описаны сообщения (т. е. структуры данных и типы данных) на основе XML-схемы, что позволяет использовать XML с учетом типов и обеспечивает упрощенную оценку действительности сообщений, которые являются объектом обмена.

[V2G2-097] При передаче сообщений V2G, описанных в настоящем стандарте, с использованием XML все субъекты V2G должны использовать формат кодирования в соответствии с определениями в *[40]*.

7.9.1.2 Эффективный XML-обмен

Формат эффективного XML-обмена (EXI) позволяет использовать и обрабатывать сообщения на базе XML на уровне двоичных кодов. Таким образом, формат EXI увеличивает скорость обработки данных на базе XML, а также сокращает использование памяти. EXI является рекомендацией W3C. Формат EXI использует для кодирования относительно простой подход на основе грамматики, обеспечивающий очень эффективное кодирование для широкого ряда случаев применения. Нередко EXI-сообщения бывают до 100 раз меньше, чем эквивалентные XML-документы. Спецификация EXI описывает с помощью предопределенного процесса, как информация схемы должна трансформироваться в грамматику EXI. Основанием для этого является то, что грамматику EXI гораздо легче обрабатывать по сравнению с информацией XML-схемы. При этом грамматический разбор может быть осуществлен так же точно, как и в XML.

Существуют разные виды механизма кодирования с EXI. Для удовлетворения требований настоящего стандарта в плане эффективности обработки, использования меньших ресурсов, размера сообщения и расширяемости сообщения должны быть выбраны настройки, учитывающие схему (см. требования к настройкам EXI для настоящего стандарта в 7.9.1.3).

EXI-потоки могут быть созданы очень эффективным способом, если вся кодированная информация (элементы/атрибуты) определена XML-схемой. Информация с отклонениями на основе знания XML-схемы кодируется более общим способом. Кодер EXI кодирует классифицированные имена (область имен и имя элемента/атрибута) неизвестной информации на базе операций со строками. Однако простые типы отклонений схемы могут по-прежнему кодироваться с учетом типа.

Декодеры EXI способны декодировать эффективные EXI-потоки, используя ту же лежащую в основе XML-схему, которая была использована для процесса кодирования. Отклонения от схемы определяются в потоке EXI. Данные отклонения (неизвестные элементы или атрибуты) могут быть обработаны или пропущены.

На рисунке 11 обобщается концепция эффективного XML-обмена для домена. Вследствие больших ограничений, связанных с лимитом ресурсов, EVCC может быть способен только обрабатывать данные на базе XML, используя соответствующее представление структуры данных. Такие структуры данных могут быть использованы для сереализации или десереализации сообщений приложения. При этом SECC может быть способен обрабатывать данные как структуру данных и/или как требующую более интенсивной работы ресурсов объектную модель документов (ОМД) или как традиционный вариант простого текста XML.

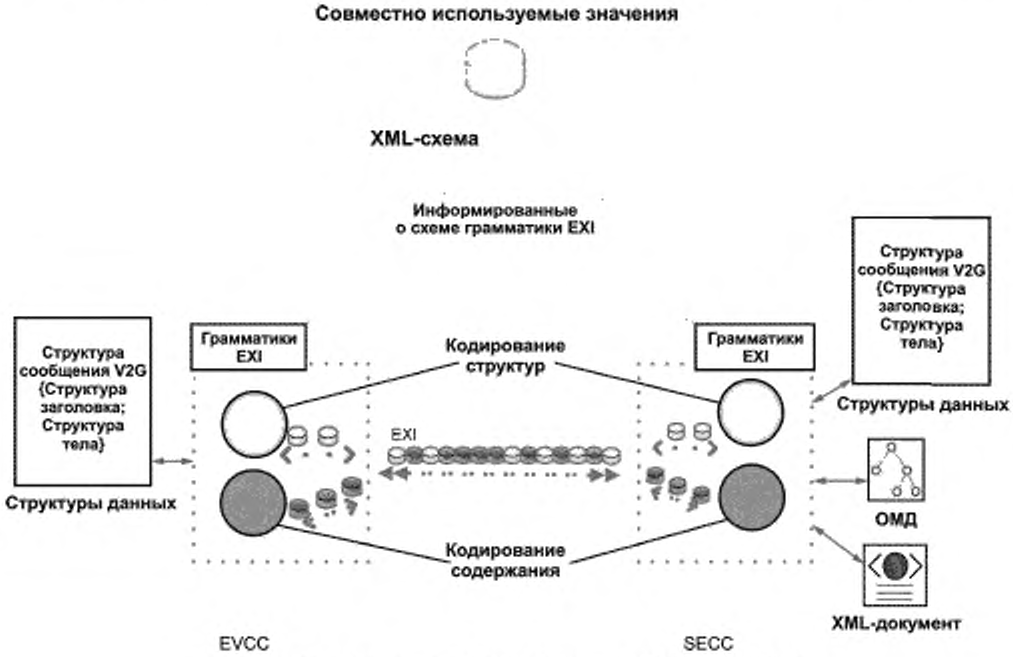


Рисунок 11 — Применение базовой концепции EXI к связи V2G

7.9.1.3 Настройки EXI для сообщений прикладного уровня

Следующие настройки EXI используются для связи V2G на базе EXI:

[V2G2-098] XML-схема с областью имен «urn:iso:15118:2:2013:msgDef», которая представляет собой версию ISO 15118 2.0 (основной номер версии «2», дополнительный — «0»), должна использоваться для кодирования и декодирования EXI-потоков.

[V2G2-099] EXI-кодер для кодирования и декодирования коммуникации в соответствии с *[1], ГОСТ Р 58122 (ИСО 15118-1)* и настоящим стандартом должен использовать опции кодирования EXI по умолчанию в соответствии с *[40]* (см. пункт «EXI Options»), за исключением опций, перечисленных в таблице 11.

Таблица 11 — Настройки опций EXI

Опция EXI	Описание	Значение *ГОСТ Р (ИСО15118-1.2013)*
valuePartitionCapacity	Указывает допустимый диапазон числа разделов в таблице строк	0

[V2G2-100] EXI-заголовок (см. *[40]*, пункт «Header») должен использоваться способом, отвечающим требованиям *[1], ГОСТ Р 58122 (ИСО 15118-1)* и настоящего стандарта. Это означает, что факультативный EXI Cookie-файл ($EXI) никогда не должен использоваться и бит присутствия для опций EXI должен быть всегда установлен в 0 (=false). Как следствие, факультативные опции EXI никогда не должны быть частью сообщения *[1], ГОСТ Р 58122 (ИСО 15118-1)* и настоящего стандарта. Каждая реализация EXI (на EVCC или SECC) должна отбрасывать сообщения, которые не соблюдают опции заголовка EXI *[1], ГОСТ Р 58122 (ИСО 15118-1)* и настоящего стандарта.

[V2G2-101] Элемент/атрибут, который не определен в области имен «urn:iso:15118:2:2013:msgDef», должен кодироваться и декодироваться как случай отклонения схемы в соответствии с *[40]* (пункт «Adding Productions when Strict is False»).

[V2G2-600] EXI-кодер для кодирования и декодирования коммуникации по *[1], ГОСТ Р 58122 (ИСО 15118-1)* и настоящему стандарту должен использовать настройки профиля EXI в соответствии с таблицей 12.

Примечание — Дополнительное описание профиля EXI см. в *[42]*.

Таблица 12 — Настройки профиля EXI

Параметр профиля EXI	Описание	Значение по *ГОСТ Р 58122 (ИСО15118-1:2013)*
maximumNumberOfBuiltInElementGrammars	Эта опция является максимальным числом встроенных грамматик элементов, для которых динамически были добавлены продукции, не являющиеся продукциями thAT(xsi:type)	0
maximumNumberOfBuiltInProductions	Эта опция является максимальным числом продукций верхнего уровня, которые могут быть динамически вставлены во встроенные грамматики элементов, за исключением продукций AT(xsi:type)	0

[V2G2-102] Простой тип/значение элемента/атрибута, который не определен в области имен «urn:iso:15118:2:2013:msgDef», должен кодироваться и декодироваться с учетом типа.

7.9.2 Безопасность сообщений

7.9.2.1 Обзор

XML-подпись является рекомендацией *[41]*, которая направлена на удовлетворение требований аутентичности некоторых фрагментов данных (например, информация об учете) V2G на базе XML. XML-подпись определяет механизм, с помощью которого сообщения и части сообщений могут быть

подписаны цифровой подписью для проведения аутентификации, обеспечения сохранности, исключения искажений. Для защиты конфиденциальности применяется схема гибридного шифрования на базе протокола Диффи — Хеллмана.

7.9.2.2 Удостоверения и наборы шифров безопасности прикладного уровня

Удостоверения, применяемые на уровне приложения, должны быть нацелены на безопасность XML. XML-подпись выбрана для защиты информации, относящейся к расчетам между EVCC, SECC и/или SA. Двоичное шифрование обеспечивает способ предоставления закрытого ключа EVCC с защитой конфиденциальности и без доступа посредников к данному закрытому ключу. Оба подхода требуют ассиметричного материала ключей. Удостоверения для подписания EVCC обеспечиваются сертификатом контракта, удостоверения для получения EVCC зашифрованных данных обеспечиваются обменом ключей ECDH, как описано в приложении D.

[V2G2-103] Максимальный срок действия сертификата контракта должен быть не более двух лет.

[V2G2-104] Минимальный срок действия сертификата, используемого для XML-подписи и обеспечения механизма для шифрования, должен быть четыре недели. Если срок действия контракта меньше, срок действия сертификата должен быть адаптирован к сроку действия контракта.

Примечание 1 — Пояснения относительно срока действия сертификата см. в С.1 (приложение С).

Примечание 2 — Если сертификат не используется для зарядки, он может быть использован для сервиса. Сервисный сертификат изготовителя позволяет EVCC запросить действительный сертификат контракта для режима PnC.

Примечание 3 — Закрытые ключи сертификата контракта и сервисного сертификата изготовителя являются данными, требующими защиты. После того как закрытый ключ взломан, ему нельзя больше доверять и поэтому им больше нельзя пользоваться. Если один из этих закрытых ключей может быть прочтен злоумышленником, то злоумышленник может выдать себя за исходный EVCC и получить электроэнергию за счет настоящего владельца транспортного средства. В качестве контрмеры EVCC может применить дополнительную защиту указанных ключей. Это может варьироваться от простых мер безопасности, таких как настройка предохранительных битов защиты от считывания в микроконтроллере EVCC, до более продвинутой защиты за счет средств защищенной загрузки, защиты при доступе для отладки и встроенных аппаратных модулей безопасности (АМБ). Последние обеспечивают высокий уровень безопасности, включая защиту против физических атак. Надлежащий АМБ обеспечивает безопасную среду для операций с закрытым ключом. Он предлагает интерфейсам использовать закрытый ключ, но обеспечивает невозможность считывания или манипулирования ключом даже для программы, которая работает в EVCC. Специализированный АМБ может даже допускать ввод зашифрованного закрытого ключа сертификата контракта с тем, чтобы закрытый ключ никогда не был доступен вне АМБ в простой незашифрованной форме. Такой АМБ может быть интегрирован в EVCC как отдельный физический модуль или как расширение кристалла микроконтроллера.

Примечание 4 — Для сокращения числа требующихся запросов об обновлении сертификата для EV оператору услуг для EV, выдающему сертификаты контрактов на зарядку, предлагается выбирать более длительные сроки действия (до максимальных) сертификата, если это имеет смысл в плане срока действия контракта и периода аннуляции.

7.9.2.3 Сертификаты контракта как удостоверения XML-подписи

Сертификат контракта привязан к EMAID и используется в XML-подписи для авторизации зарядки транспортного средства. Сертификат контракта может быть проверен, даже если SECC находится в режиме офлайн. Связывание контракта осуществляется следующим образом:

[V2G2-108] EMAID должен кодироваться в предмете сертификата.

7.9.2.4 Специфика безопасности XML для набора(ов) сообщений PnC

7.9.2.4.1 Структуры XML-данных для безопасности прикладного уровня

Безопасность на уровне приложения обеспечивается с помощью подписи и шифрования сообщений. Обмен информацией, предназначаемой для сервисов SA, осуществляется с использованием структур данных XML. Поэтому эта информация может быть целиком защищена с помощью средств безопасности XML.

Типичная информация, которой обмениваются EVCC и SECC, это согласование подписанных показаний прибора учета от транспортного средства, используемых для онлайновых и полуонлайновых соединений. Показания прибора учета от SECC отправляются через защищенный канал TLS. Они могут включать подпись счетчика электроэнергии, обеспечивающую аутентификацию

источника для дополнительной защиты. Транспортное средство, в свою очередь, подписывает показания прибора учета для поддержки процесса расчета, если местные правила это допускают. Такой подход позволяет экономить подписи показаний прибора учета со стороны SECC. Показания прибора учета накапливаются, поэтому самое последнее показание прибора учета служит основой для расчета. Используются XML-подписи, которые обеспечивают защиту сохранности информации, давая возможность всем промежуточным и участвующим субъектам полагаться на эту информацию.

7.9.2.4.2 Механизм XML-подписи

В данном пункте описан механизм XML-подписи.

Примечание 1 — Обзор XML-подписей см. также в приложении Е.

XML-подписи, которые соответствуют синтаксису и версии обработки 1.1 XML-подписи *[41]*, могут быть применены к произвольному цифровому содержанию (объектам данных) так же, как и при расчете цифровых подписей. При применении цифровой подписи к объектам данных они сначала обрабатываются (хешируются), и результат затем подписывается с помощью асимметричного алгоритма, такого как RSA (алгоритм цифровой подписи Ривеста — Шамира — Адельмана) или ECDSA. В случае XML свертка помещается в XML-элемент вместе с дополнительной информацией. Этот элемент затем хешируется и криптографически подписывается. В настоящем стандарте используются обособленные XML-подписи, которые соответствуют синтаксису и версии обработки 1.1 XML-подписи (см. *[41]*). Это означает, что подпись и данные могут быть в отдельном (внешне обособленные) или в одном и том же XML-документе (внутренне обособленные) как родственные элементы одного уровня. Подпись может охватывать только часть XML-документа, на который ссылается ObjectID.

На рисунке 12 показана диаграмма схемы элемента XML-подписи, включенного в заголовок сообщения V2G.

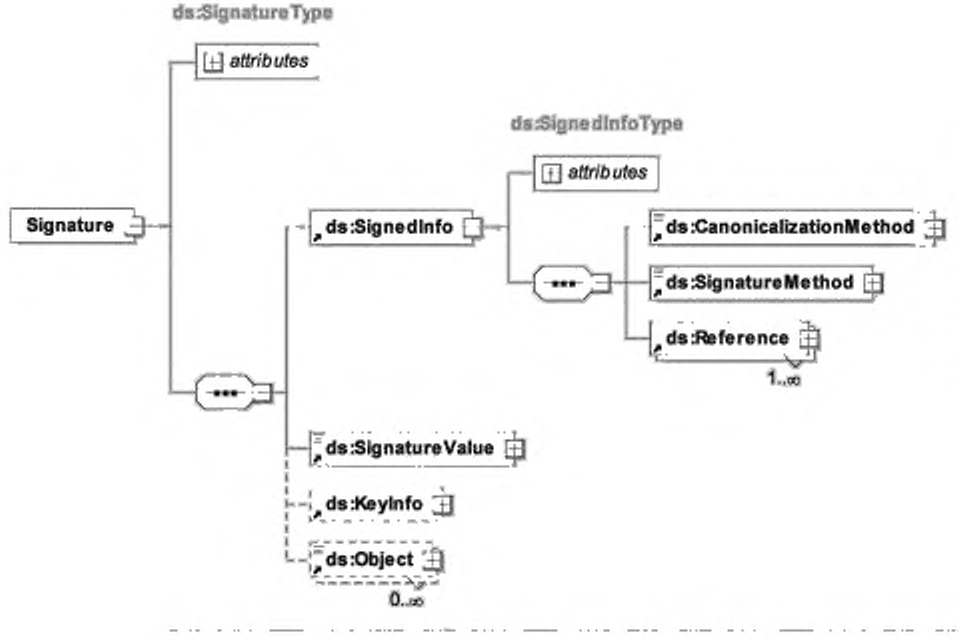


Рисунок 12 — Диаграмма — XML-подпись

На рисунке 13 показана диаграмма элемента Reference, включенного в элемент SignedInfo, который является частью XML-подписи, показанной на рисунке 12.

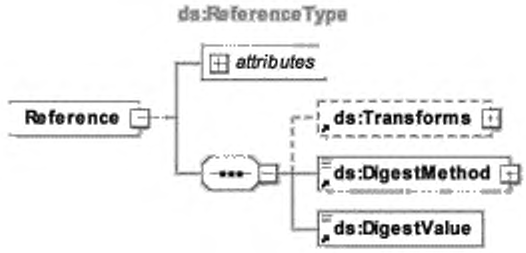


Рисунок 13 — Диаграмма — Элемент Reference, включенный в элемент SignedInfo

Примечание 2 — Значения прибора учета от SECC могут быть уже подписанными. Это значение подписи рассматривается как информационный элемент и требуется для процесса измерений. Подпись может также включать идентификатор прибора учета.

[V2G2-117] Каждый субъект V2G должен поддерживать обособленные XML-подписи.

[V2G2-119] Для операций XML-подписей подписываемые данные должны быть EXI-представлением этих данных.

[V2G2-764] Как метод канонизации должен использоваться EXI с грамматикой фрагмента, учитывающей схему.

[V2G2-765] Каждое сообщение, требующее структуры XML-подписи, должно использовать значение «http://www.w3.org/TR/canonical-exi/» в качестве атрибута алгоритма в элементе CanonicalizationMethod.

[V2G2-766] Каждое сообщение, требующее структуры XML-подписи, должно использовать значение «http://www.w3.org/TR/canonical-exi/» в качестве атрибута алгоритма в элементе Transform.

[V2G2-767] Максимальное число алгоритмов Transform ограничивается одним (1) (т. е. на один элемент, на который делается ссылка и для которого передается подпись, может быть указан только один алгоритм Transform).

[V2G2-768] Части, которые должны быть подписаны в сообщениях на базе XML, должны кодироваться как фрагмент, учитывающий EXI-схему.

[V2G2-769] Каждое сообщение, требующее структуры XML-подписи, должно использовать значение «http://www.w3.org/2001/04/xmldsig-more#ECDSA-sha256» в качестве атрибута алгоритма в элементе SignatureMethod.

[V2G2-770] Каждое сообщение, требующее структуры XML-подписи, должно использовать значение «http://www.w3.org/2001/04/xmlenc#sha256» в качестве атрибута алгоритма в элементе DigestMethod.

[V2G2-771] Следующие элементы сообщений со структурой XML-подписи не должны использоваться при передаче подписей в заголовке сообщения V2G:
- Id (attribute in SignedInfo);
- ##any in SignedInfo — CanonicalizationMethod;
- HMACOutputLength in SignedInfo — SignatureMethod;
- ##other in SignedInfo — SignatureMethod;
- Type (attribute in SignedInfo-Reference);
- ##other in SignedInfo — Reference — Transforms — Transform;
- XPath in SignedInfo — Reference — Transforms — Transform;
- ##other in SignedInfo — Reference — DigestMethod;
- Id (attribute in SignatureValue);
- Object (in Signature);
- KeyInfo.

[V2G2-909] Подпись не должна относиться к более чем четырем подписываемым элементам.

Примечание 3 — Это позволяет определить верхнюю границу для размера заголовка подписи.

Для применения XMLDsig любой элемент, который должен быть подписан, должен быть адресуемым. В настоящем стандарте это обеспечивается ссылкой универсального индикатора ресурсов (URI) на идентификационный атрибут такого элемента. Таким образом, любой подписанный элемент сообщения несет идентификационный атрибут. Если конкретный элемент должен подписываться во всех случаях использования, то идентификационный атрибут обязательно помечается в XSD. Однако если он подписывается только в некоторых случаях использования, то идентификационный атрибут помечается как факультативный и может быть опущен, когда он не требуется.

Примечание 4 — Присутствие идентификационного атрибута не обязательно означает, что подпись используется, т. е. если подпись не используется, идентификатор может тем не менее присутствовать.

7.9.2.4.3 Механизм шифрования

Закрытые ключи, принадлежащие сертификатам контрактов, требуют защиты (т. е. шифрования) при передаче от SA к EVCC. Более подробные пояснения приведены в приложении D.

[V2G2-121] EVCC должен поддерживать расчет секретной функции и функции установления ключа ECDH для расшифровки зашифрованной информации, такой как закрытые ключи.

[V2G2-122] Каждый субъект V2G должен иметь механизмы для выполнения обмена ключами ECDH. Открытые параметры являются производными от открытых параметров ECDSA.

[V2G2-814] Закрытый ключ, соответствующий сертификату контракта, должен передаваться только в зашифрованном формате в сообщениях CertificateInstallationRes и CertificateUpdateRes — в составе ContractSignatureEncryptedPrivateKey.

[V2G2-815] Закрытый ключ, соответствующий сертификату контракта, должен шифроваться отправителем (SA) с использованием ключа сеанса, полученного в протоколе ECDH (см. **[V2G2-818]**), и алгоритма AES-CBC-128 в соответствии с *[43]*. Вектор инициализации IV должен случайно генерироваться перед шифрованием, иметь длину 128 бит и никогда повторно не использоваться. Вектор инициализации IV должен передаваться в 16 старших байтах поля ContractSignatureEncryptedPrivateKey.

Примечание 1 — Подразумевается, что надлежащее случайное генерирование векторов инициализации IV обеспечивает исключение их повторного использования с чрезвычайно высокой вероятностью, эквивалентной достоверности для криптографических требований.

[V2G2-816] Порядок байтов закрытого ключа должен быть обратным, включая ведущие нули.

Примечание 2 — Поскольку используется ECDSA с кривой secp256r1, закрытый ключ имеет длину 256 бит.

Примечание 3 — Дополнение нешифрованного текста (закрытый ключ) не требуется, поскольку его длина кратна размеру блока используемого алгоритма шифрования.

[V2G2-817] Закрытый ключ, соответствующий сертификату контракта, должен декодироваться получателем (EVCC) с помощью ключа сеанса, полученного в протоколе ECDH (см. **[V2G2-818]**), с применением алгоритма AES-CBC-128 в соответствии с *[43]*. Вектор инициализации IV должен считываться из 16 старших байтов поля ContractSignatureEncryptedPrivateKey.

[V2G2-818] Для согласования ключа сеанса должен использоваться эфемерно-статический протокол публикации *[44]*. KDF должен быть «конканатенационным KDF», в котором функция хеширования должна быть SHA256. SA должен действовать как сторона U (в соответствии с *[44]*), и EVCC должен действовать как сторона V. Протокол должен использовать эллиптические кривые, как указано в приложении B. Идентификатор алгоритма должен быть обозначен одним символом 0x01. Имя отправителя IDU должно быть обозначено одним символом «U» = 0x55, имя получателя IDV — одним символом «V» = 0x56. Должен быть получен симметричный ключ шифрования — точно 128 бит.

Примечание 4 — Аутентичность передачи обеспечивается окружающей подписью. Проверка аутентичности обязательна для безопасности протокола ECDH.

[V2G2-819] Эфемерные открытые ключи должны кодироваться как «Subject Public Key» в соответствии с *[45]* и содержаться в XML-элементе «DHpublickey». Должна использоваться исключительно несжатая форма.

Примечание 5 — XML-элемент «DHpublickey» имеет длину 65 байтов. Первый байт имеет фиксированное значение 0x04, что указывает на несжатую форму.

[V2G2-820] В сообщении CertificateInstallationRes открытый ключ получателя QV должен быть открытым ключом, содержащимся в существующем сервисном сертификате изготовителя.

[V2G2-821] В сообщении CertificateUpdateRes открытый ключ получателя QV должен быть открытым ключом, содержащимся в существующем сертификате контракта.

[V2G2-822] Сертификат, пара ключей которого используется для ECDH, должен иметь установленные в единицу флаги использования ключей keyAgreement. Это требование действует для всех сертификатов контракта и всех сервисных сертификатов изготовителя и должно выполняться всеми участвующими сторонами.

[V2G2-823] После получения сертификата контракта EVCC должен убедиться в том, что закрытый ключ, полученный с сертификатом, является действительным закрытым ключом для этого сертификата: его значение должно быть строго меньше, чем порядок базовой точки, и умножение базовой точки на это значение должно генерировать ключ, который соответствует открытому ключу сертификата контракта.

7.9.2.4.4 Генерирование случайных чисел

Алгоритм ECDSA в большей степени полагается на тот факт, что злоумышленник не имеет информации о случайном числе. В противном случае подписи будут допускать утечку информации о секретном ключе. Несовершенство реализации генератора случайных чисел может позволить злоумышленнику извлечь секретный ключ после прочтения определенного числа подписей. В худшем случае простое повторное использование случайного числа позволит злоумышленнику рассчитать закрытый ключ после прочтения только двух подписей. Поэтому чрезвычайно важно иметь надлежащий генератор случайных чисел (ГСЧ), так называемый криптографически безопасный ГСЧ. Кроме того, криптографически безопасный ГСЧ необходим для генерирования вектора инициализации в AES-CBC и для генерирования вызова.

[V2G2-835] Каждый раз, когда в настоящем стандарте субъект V2G требует случайных чисел, должен использоваться современный криптографически безопасный генератор случайных чисел.

Примечание — В реализации может присутствовать подходящий криптографически безопасный генератор случайных чисел. Поскольку фактическая реализация не влияет на функциональную совместимость, настоящий стандарт не предписывает конкретных мер, чтобы соответствовать требованиям завтрашнего дня и избежать дублирования. Однако установленные стандарты должны соблюдаться, например *[46]* и *[47]*.

7.9.2.4.5 Применение механизмов безопасности к XML-сообщению

Поддерживаются две пары механизмов безопасности:

- аутентичность и сохранность: Генерирование подписи. Проверка подписи.

Применяется подпись на базе XML. Субъект, который создает XML-сообщение, подписывает определенные или все поля XML-сообщения. Получатель проверяет подпись;

- конфиденциальность: Шифрование. Дешифрование.

Применяется асимметричное шифрование. Субъект, который создает сообщение, шифрует единое двоичное поле XML-сообщения. Получатель расшифровывает это двоичное поле.

В таблицах 13 и 14 дается обзор применяемых механизмов безопасности.

Таблица 13 — Обзор применяемых подписей на основе XML

XML-сообщение	Защищаемые поля	Подписывающий субъект (отправитель)	Проверяющий субъект (получатель)
AuthorizationReq	Тело сообщения/Все поля (если присутствует GenChallenge)	EVCC	SECC
CertificateUpdateReq	Тело сообщения/Все поля	EVCC; подписывается сертификатом контракта, цепочка передается в элементе тела сообщения	Вторичный субъект

Окончание таблицы 13

XML-сообщение	Защищаемые поля	Подписывающий субъект (отправитель)	Проверяющий субъект (получатель)
CertificateUpdateRes	ContractSignatureCertChain ContractSignatureEncryptedPrivateKey DHpublickey eMAID	Вторичный субъект: служба выдачи сертификатов	EVCC; цепочка сертификата сохраняется после успешной проверки
CertificateInstallationReq	Тело сообщения/Все поля	EVCC; подписывается сертификатом изготовителя, передается в элементе тела сообщения	Вторичный субъект
CertificateInstallationRes	ContractSignatureCertChain ContractSignatureEncryptedPrivateKey DHpublickey ContractID	Вторичный субъект: служба выдачи сертификатов	EVCC; цепочка сертификата сохраняется после успешной проверки
MeteringReceiptReq	Тело сообщения/Все поля (дополнительно)	EVCC	SECC
ChargeParameterDiscoveryRes	SalesTariff (дополнительно, требуется для PnC)	Вторичный субъект: оператор услуг EV Sub-CA 2	EVCC

Таблица 14 — Обзор шифрования на уровне приложения

XML-сообщение	Защищаемые поля	Шифрующий субъект (отправитель)	Дешифрующий субъект (получатель)
CertificateUpdateRes	ContractSignatureEncryptedPrivateKey	Вторичный субъект	EVCC
CertificateInstallationRes	ContractSignatureEncryptedPrivateKey	Вторичный субъект	EVCC

[V2G2-652] Каждый субъект V2G должен быть способен генерировать XML-подписи, как указано в таблице 13.

[V2G2-653] Каждый субъект V2G должен быть способен проверять XML-подписи, как указано в таблице 13.

7.9.2.5 Выдача сертификатов

Оператор услуг для EV создает удостоверения для EVCC, состоящие из сертификата контракта, закрытого ключа, который был асимметрично зашифрован специально для данного EVCC. Затем оператор услуг для EV направляет удостоверения службе выдачи сертификатов. Сервис выдачи сертификатов подтверждает корректность и аутентичность удостоверений своей подписью и передает подписанный фрагмент сообщения SECC, который составляет сообщение CertificateInstallationRes или CertificateUpdateRes и передает его EVCC. Служба выдачи сертификатов считается надежной, поэтому не требуется, чтобы EVCC был способен проверять сертификат, который он получил.

На рисунке 14 приведен процесс для CertificateInstallationRes и CertificateUpdateRes.

Роль «службы выдачи сертификатов» может быть принята на себя, например, самим оператором услуг для EV, оператором SECC, полностью автономной службой или (при условии тщательного анализа возможных последствий в отношении безопасности) даже самим EVCC.

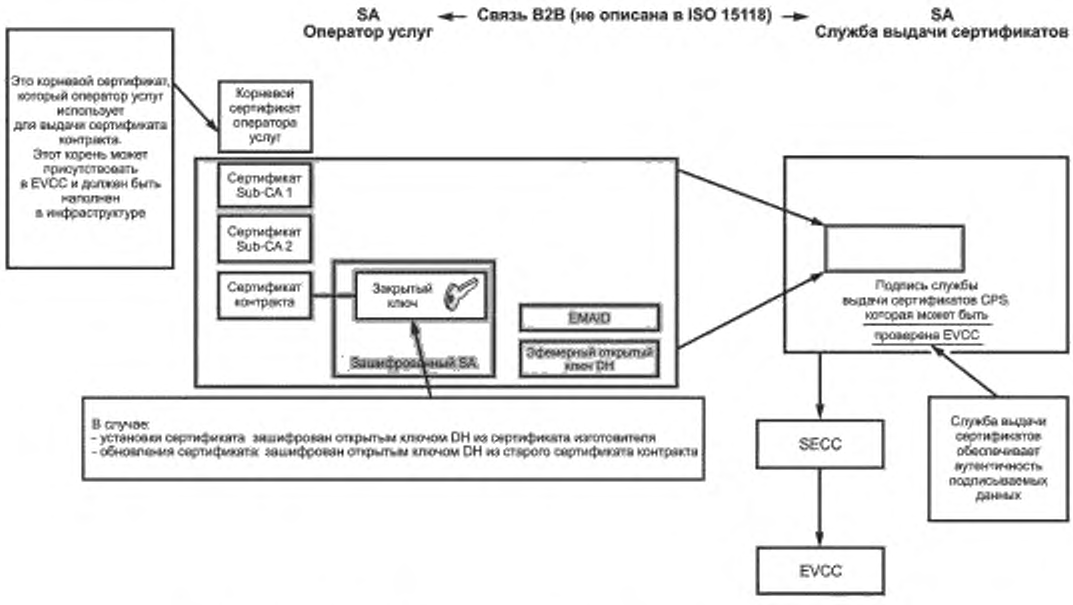


Рисунок 14 — Процесс для CertificateInstallationRes и CertificateUpdateRes

7.10 Уровень приложения

7.10.1 Протокол обнаружения SECC

7.10.1.1 Общая информация

EVCC использует протокол обнаружения SECC (SDP) для получения IP-адреса и номера порта SECC. Клиент SDP посылает сообщения с запросом об обнаружении SECC по локальному каналу (групповой адрес), ожидая что какой-либо сервер SDP ответит на его запрос ответным сообщением об обнаружении SECC, содержащим эту информацию.

После того как EVCC получил IP-адрес и номер порта SECC, он может установить соединение транспортного уровня с SECC (см. 7.3.4).

[V2G2-123] Сервер SDP должен быть доступен по локальному каналу.

Примечание — Как это часто встречается в интернет-технологиях, сервер SDP может быть реализован на одном физическом устройстве с SECC и может быть связанным с тем же самым IP-адресом. Если это не тот случай, то оптимистическое определение дублирования адресов, описанное в *[19]*, не приведет к преимуществам.

7.10.1.2 Поддерживаемые порты

SDP — это протокол на основе UDP. Порты, перечисленные в таблице 15, используются SDP.

Таблица 15 — Поддерживаемые порты UDP для SDP

Имя	Протокол	Номер порта	Описание
V2G_UDP_SDP_CLIENT	UDP (адресация конкретному устройству)	Номер порта в диапазоне динамических портов (49152-65535) в соответствии с *[39]*	Порт источника клиента SDP на EVCC
V2G_UDP_SDP_SERVER	UDP (групповая адресация)	По ГОСТ Р 58123	Порт сервера SDP, который принимает пакеты UDP с групповым локальным IP-адресом назначения

[V2G2-124] Клиент SDP должен поддерживать порт V2G_UDP_SDP_CLIENT для отправления и получения сообщений SDP, как указано в таблице 15.

[V2G2-125] Сервер SDP должен поддерживать порт V2G_UDP_SDP_SERVER для получения и отправления сообщений SDP, как указано в таблице 15.

Примечание — В зависимости от реализации EVCC динамически назначаемый порт V2G_UDP_SDP_CLIENT будет назначен однажды во время или перед первой передачей пакета UDP SECC или может быть динамически переназначен для каждого индивидуального сообщения-запроса UDP и ответа. Также в зависимости от того, посылаются ли сообщения повторно, сообщения-ответы могут приходить асинхронно и уже не могут быть ассоциированы с конкретным соответствующим запросом.

[V2G2-126] Клиент SDP должен быть способен обрабатывать асинхронно получаемые сообщения с ответом об обнаружении SECC.

7.10.1.3 Блок протокольных данных

7.10.1.3.1 Структура

Сообщение SDP базируется на формате сообщения V2GTP, как указано в 7.8.3.1.

[V2G2-127] Клиент SDP должен поддерживать определения 7.8.3.1.

[V2G2-128] Клиент SDP должен использовать отдельный пакет UDP для каждого сообщения с запросом.

[V2G2-129] Клиент SDP должен поместить первый байт заголовка сообщения с запросом, показанного на рисунке 9 и в таблице 9, в первом байте полезных данных пакета UDP.

[V2G2-130] Сервер SDP должен поддерживать определения 7.8.3.1.

[V2G2-131] Сервер SDP должен использовать отдельный пакет UDP для каждого ответа.

[V2G2-132] Сервер SDP должен поместить первый байт заголовка сообщения-ответа, показанного на рисунке 9 и в таблице 9, в первом байте полезных данных пакета UDP.

7.10.1.3.2 Обработка заголовка

Обработка заголовка SDP базируется на обработке заголовка сообщения V2GTP, как указано в 7.8.3.2.

[V2G2-133] Клиент SDP должен применяться для обработки заголовка, как указано в 7.8.3.2.

[V2G2-134] Сервер SDP должен применяться для обработки заголовка, как указано в 7.8.3.2.

7.10.1.4 Сообщение-запрос об обнаружении SECC

Клиент SDP использует сообщение-запрос об обнаружении SECC для запроса IP-адреса и номера порта SECC.

[V2G2-135] Только клиент SDP должен отправлять сообщения-запросы об обнаружении SECC.

[V2G2-136] Клиент SDP должен отправлять сообщения-запросы об обнаружении SECC с IP-адресом источника, по которому он ожидает сообщение с ответом об обнаружении SECC.

[V2G2-137] Клиент SDP должен отправлять сообщения-запросы об обнаружении SECC на порт назначения V2G_UDP_SDP_SERVER, как указано в таблице 15.

[V2G2-138] Клиент SDP должен отправлять сообщения-запросы об обнаружении SECC с указанием на порт-источник V2G_UDP_SDP_CLIENT, как указано в таблице 15, на который он ожидает сообщение-ответ об обнаружении SECC.

[V2G2-139] Клиент SDP должен отправлять сообщение-запрос об обнаружении SECC на групповой локальный адрес канала назначения (FF02::1), как описано в *[22]*.

[V2G2-140] Клиент SDP должен отправлять сообщение-запрос об обнаружении SECC со значением 0x9000 типа полезных данных, как указано в таблице 10.

[V2G2-141] Клиент SDP должен отправлять сообщение-запрос об обнаружении SECC с длиной 2 полезных данных.

[V2G2-142] Клиент SDP должен отправлять сообщение-запрос об обнаружении SECC с полезными данными, показанными на рисунке 15.

[V2G2-622] Клиент SDP должен отправлять полезные данные в порядке, показанном на рисунке 15. Байт с меньшим номером должен отправляться перед байтом с большим номером. Полезные данные начинаются с байта 1 и заканчиваются байтом 2.

[V2G2-623] Клиент SDP должен использовать кодирование для запрашиваемой опции безопасности и запрашиваемого транспортного протокола, как указано в таблице 16.

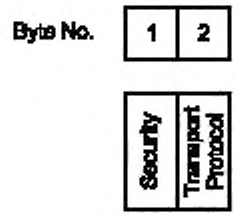


Рисунок 15 — Полезные данные сообщения-запроса об обнаружении SECC

Таблица 16 — Кодирование безопасности SDP и опций протокола

Описание	Безопасность	Транспортный протокол
№ байта сообщения-запроса SDP	1	2
№ байта сообщения-ответа SDP	19	20
Применяемые значения	0x00 = защищено TLS 0x01-0x0F = зарезервировано 0x10 = нет безопасности транспортного уровня 0x11-0xFF = зарезервировано	0x00= TCP 0x01-0x0F = зарезервировано 0x10 = зарезервировано для TCP 0x11-0xFF = зарезервировано

7.10.1.5 Сообщение-ответ об обнаружении SECC

Сервер SDP использует сообщение-ответ об обнаружении SECC, чтобы ответить на сообщение-запрос об обнаружении SECC и предоставить клиенту IP-адреса и порта SECC.

[V2G2-143] Сервер SDP должен быть способен извлечь IP-адрес источника и номер порта источника полученного пакета UDP (IP-адрес и номер порта клиента) и отправить пакет UDP на идентифицированный IP-адрес и номер порта.

[V2G2-144] Сервер SDP должен ответить на любое сообщение-запрос об обнаружении SECC сообщением-ответом об его обнаружении.

Примечание 1 — Это требование обеспечивает возможность доступа к серверу SDP, обслуживающему несколько клиентов, в любое время. Это поддерживает зарядку нескольких EV на EVSE одним SECC.

[V2G2-145] Клиент SDP не должен отвечать на любое сообщение-запрос об обнаружении SECC.

[V2G2-146] Сервер SDP должен отправлять сообщения-ответы только после получения сообщения с запросом об обнаружении SECC.

[V2G2-147] Сервер SDP должен отправлять сообщения-ответы об обнаружении SECC только после получения сообщения-запроса об обнаружении SECC.

Примечание 2 — Подробные требования по хронированию см. в требованиях **[V2G2-159]—[V2G2-162]**.

[V2G2-149] Сервер SDP должен отправить ответ SDP с портом источника V2G_UDP_SDP_SERVER, как указано в таблице 15.

[V2G2-150] Сервер SDP должен отправлять сообщение-ответ об обнаружении SECC клиенту SDP, который отправил сообщение-запрос об обнаружении SECC.

[V2G2-151] Сервер SDP должен отправлять сообщение-ответ об обнаружении SECC на порт клиента SDP, который отправил сообщение-запрос об обнаружении SECC.

[V2G2-152] Сервер SDP должен отправлять сообщение-ответ об обнаружении SECC со значением 0x9001 типа полезных данных, как указано в таблице 10.

[V2G2-153] Сервер SDP должен отправлять сообщение-ответ об обнаружении SECC с длиной полезных данных, равной 20.

[V2G2-154] Сервер SDP должен отправлять сообщение-ответ об обнаружении SECC с полезными данными, показанными на рисунке 16.

[V2G2-155] Сервер SDP должен отправлять полезные данные в порядке, показанном на рисунке 16. Байт с меньшим номером должен отправляться перед байтом с большим номером. Полезные данные начинаются с байта 1 и заканчиваются байтом 20.

[V2G2-156] Сервер SDP должен отправлять поля «SECC IP Address» и «SECC Port» в формате с обратным порядком следования байтов: старший байт передается первым, младший байт — последним.

Примечание 3 — Механизм, используемый сервером SDP для определения своего собственного IP-адреса, не рассматривается в настоящем стандарте.

Примечание 4 — IP-адрес источника и порт источника полученного пакета UDP обычно предоставляются стеком TCP/IP.

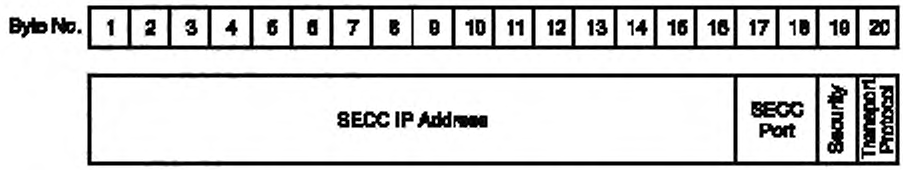


Рисунок 16 — Полезные данные сообщения об обнаружении SECC

[V2G2-624] Для определения поддерживаемого протокола безопасности передачи и транспортного протокола для указанного порта сервер SDP должен использовать кодирование соответственно запрашиваемой опции безопасности и запрашиваемого транспортного протокола, как указано в таблице 16, предоставленных в тех же полезных данных, что и байты протокола безопасности и транспортного протокола.

7.10.1.6 Хронирование и обработка ошибок

Процесс обнаружения SECC основан на определениях тайм-аута из приложения, как описано в 7.10.1. В этом пункте описываются дополнительное хронирование и обработка ошибок для протокола обнаружения SECC.

[V2G2-157] Клиент SDP должен подсчитывать число сообщений-запросов об обнаружении SECC до получения действительного сообщения-ответа об обнаружении SECC.

[V2G2-158] Клиент SDP должен сбросить счетчик отправленных сообщений-запросов об обнаружении SECC после получения действительного сообщения-ответа об обнаружении SECC.

[V2G2-159] После отправления сообщения-запроса об обнаружении SECC клиент SDP должен ожидать сообщения-ответа об обнаружении SECC не менее 250 мс.

[V2G2-160] После безуспешного ожидания сообщения-ответа об обнаружении SECC клиент SDP должен отправить новое сообщение-запрос об обнаружении SECC и инкрементировать счетчик отправленных сообщений-запросов об обнаружении SECC.

[V2G2-161] Если клиент SDP не получил никакого сообщения-ответа об обнаружении SECC после последовательного отправления максимум 50 сообщений-запросов об обнаружении SECC, он должен остановить обнаружение SECC.

[V2G2-162] После прекращения обнаружения SECC клиент SDP должен перейти в такое же состояние, как и для тайм-аута (см. **[V2G2-020]** и рисунок 6).

7.10.1.7 Протокол и обработка опций безопасности

Для обработки запросов и ответов SDP клиент SDP и сервер SDP должны выполнять следующие требования:

[V2G2-625] Если клиент SDP отправляет запрос SDP с кодом опции протокола для TCP («TCP» в таблице 16), SECC должен отправить ответ SDP с кодом опции протокола «TCP».

[V2G2-626] Если клиент SDP отправляет запрос SDP с кодом опции безопасности для TLS («защищено TLS» в таблице 16) и SECC поддерживает TLS, сервер SDP должен отправить ответ SDP с кодом опции безопасности «TLS».

[V2G2-627] Если клиент SDP отправляет запрос SDP с кодом опции безопасности для TLS («защищено TLS» в таблице 16) и SECC не поддерживает TLS, сервер SDP должен отправить ответ SDP с кодом опции отсутствия безопасности «Нет безопасности транспортного уровня».

Для установления TCP/TLS действуют следующие требования:

[V2G2-628] EVCC в зависимости от случая использования и требований безопасности должен либо использовать протокол, как указано в ответе SDP от сервера SDP, либо остановить установление связи.

[V2G2-629] Если EVCC пытается осуществлять обмен данными с помощью транспортного протокола, указанного в таблице 16, который отличается от опций протокола, указанных в ответе SDP от сервера SDP, SECC не должен принимать эту коммуникацию.

[V2G2-163] Если SDP выдает глобальный IP-адрес SECC, EVCC не должен указывать обнаруженный IP-адрес SECC до тех пор, пока EVCC не сконфигурирует глобальный IP-адрес, как указано в 7.6.3.2 и 7.6.3.3.

[V2G2-164] Если SDP выдает глобальный IP-адрес SECC, EVCC должен указать обнаруженный IP-адрес SECC после присвоения глобального адреса в соответствии с 7.6.3.2 и 7.6.3.3.

7.10.2 Сообщения прикладного уровня V2G

Определения сообщений взаимодействия транспортного средства с электросетью на уровне приложения описывают обмен сообщениями типа клиент — сервер между EVCC и SECC с целью инициализации и конфигурирования процесса зарядки EV. Этот набор сообщений предназначен для случаев использования, которые описаны в *ГОСТ Р 58122 (ИСО 15118-1)*. Сообщения и необходимый поток сообщений (т. е. протокол связи) представляют собой прикладной уровень в соответствии с моделью уровневой архитектуры OSI.

Набор сообщений, поток сообщений и поведение, характерное для конкретных сообщений, описаны в разделе 8.

7.10.3 Сервисные примитивы прикладного уровня

7.10.3.1 A-Data.request

A-Data.request объявляет действия по обмену сообщениями V2G. В таблице 17 описываются сервисный примитив и его параметр(ы).

Таблица 17 — Сервисный примитив A-Data.request

Имя примитива	A-Data.request
Поддерживающий субъект	EVCC
Имя параметра	Описание
A_Msg	- Установление сеанса - Обнаружение сервиса - Параметры сервиса - Выбор сервиса и оплаты - Параметры оплаты - Авторизация зарядки - Получение информации о параметрах зарядки - Передача энергии - Статус зарядки - Учетная квитанция - Актуализация сертификата - Установка сертификата - Проверка кабеля - Предзарядка - Запрос тока - Обнаружение сварки - Остановка сеанса

A-Data.request (A_msg = «message name») запрашивает более низкий уровень для отправления сообщения-запроса V2G типа сообщения V2G, который задается параметром A_msg.

Пример — A-Data.request (A_msg = Session Setup) инициирует отправление сообщения SessionSetupReq, как указано в 8.4.3.2.1.

7.10.3.2 A-Data.indication

A-Data.indication объявляет действия по обмену сообщениями V2G. В таблице 18 описываются сервисный примитив и его параметр(ы).

Таблица 18 — Сервисный примитив A-Data.indication

Имя примитива	A-Data.indication
Поддерживающая субъект	SECC
Имя параметра	Описание
A_Msg	- Установление сеанса - Обнаружение сервиса - Параметры сервиса - Выбор сервиса и оплаты - Параметры оплаты - Авторизация зарядки - Получение информации о параметрах зарядки - Передача энергии - Статус зарядки - Учетная квитанция - Актуализация сертификата - Установка сертификата - Проверка кабеля - Предзарядка - Запрос тока - Обнаружение сварки - Остановка сеанса

A-Data.indication (A_msg = «message name») указывает на получение сообщения-запроса V2G типа сообщения V2G, который задан верхнему уровню параметром A_msg.

Пример — A-Data.indication (A_msg = Session Setup) инициирует получение сообщения SesstionSetupReq, как указано в 8.4.3.2.1.

7.10.3.3 A-Data.response

A-Data.response уведомляет о статусе обмена сообщениями V2G. В таблице 19 описываются сервисный примитив и его параметр(ы).

Таблица 19 — Сервисный примитив A-Data.response

Имя примитива	A-Data.response
Поддерживающий субъект	SECC
Имя параметра	Описание
A_msg	- Установление сеанса - Обнаружение сервиса - Параметры сервиса - Выбор сервиса и оплаты - Параметры оплаты - Авторизация зарядки - Получение информации о параметрах зарядки - Передача энергии - Статус зарядки - Учетная квитанция - Актуализация сертификата - Установка сертификата - Проверка кабеля - Предзарядка - Запрос тока - Обнаружение сварки - Остановка сеанса

A-Data.response (A_msg = «message name») указывает нижнему уровню на необходимость отправления сообщения-ответа V2G с типом сообщения V2G, который задан параметром A_msg.

Пример — A-Data.response (A_msg = Session Setup) инициирует отправление сообщения SesstionSetupRes, как указано в 8.4.3.2.2.

7.10.3.4 A-Data.confirmation

A-Data.confirmation уведомляет о статусе обмена сообщениями V2G. В таблице 20 описываются сервисный примитив и его параметр(ы).

Таблица 20 — Сервисный примитив A-Data.confirmation

Имя примитива	A-Data.confirmation
Поддерживающий субъект	EVCC
Имя параметра	Описание
A_msg	- Установление сеанса - Обнаружение сервиса - Параметры сервиса - Выбор сервиса и оплаты - Параметры оплаты - Авторизация зарядки - Получение информации о параметрах зарядки - Передача энергии - Статус зарядки - Учетная квитанция - Актуализация сертификата - Установка сертификата - Проверка кабеля - Предзарядка - Запрос тока - Обнаружение сварки - Остановка сеанса

A-Data.confirmation (A_msg = «message name») указывает на успешное получение сообщения-ответа для сообщения V2G, которое дано верхнему уровню параметром A_msg.

Пример — A-Data.confirmation (A_msg = Session Setup) указывает на успешное получение сообщения SessionSetupRes, как указано в 8.4.3.2.2.

8 Сообщения прикладного уровня

8.1 Общая информация и определения

Сообщение V2G использует представительский уровень на базе EXI, как описано в 7.9.1. Коммуникация между EVCC и SECC на прикладном уровне основана на архитектуре клиент/сервер. EVCC всегда выступает в качестве клиента (запрашивающего сервис) во время всего процесса зарядки, в то время как SECC всегда выступает в качестве обслуживающего устройства (отвечающего на запрос о сервисе). Поэтому EVCC всегда инициирует связь путем отправления сообщения-запроса SECC, который затем возвращает соответствующее сообщение-ответ. Все сообщения, обмен которыми происходит между EVCC и SECC, описываются их синтаксисом и их семантикой в 8.2, 8.3 и 8.5. Полное определение XML-схемы, описывающее оба набора сообщений V2G, содержится в приложении F.

В 8.6 описываются сообщение V2G и соответствующие элементы сообщений, поддержка которых требуется для определенного набора элементов случаев использования, приведенных в *ГОСТ 58122 (15118-1:2013)*.

В 8.7 описываются хронирование сообщений и обработка ошибок для обмена сообщениями V2G. Примеры типичных последовательностей сообщений даны в 8.8.

Коммуникация V2G состоит из двух разных наборов сообщений:

- сообщения подтверждения протокола V2G на уровне приложения (см. 8.2);
- сообщения V2G на уровне приложения (см. 8.3).

[V2G2-809] При передаче сообщений на уровне приложения должно применяться правило обратного порядка следования байтов: старший байт передается первым, младший байт — последним.

8.2 Определение подтверждения протокола

8.2.1 Последовательность подтверждения

[V2G2-165] Перед началом обмена сообщениями на уровне приложения между EVCC и SECC должен быть согласован надлежащий протокол прикладного уровня, включая его версию.

Для согласования протокола между EVCC и SECC осуществляется следующее подтверждение протокола прикладного уровня:

[V2G2-166] EVCC должен инициировать подтверждение, отправив SECC сообщение supportedAppProtocolReq, как показано на рисунке 17. Это сообщение-запрос дает перечень протоколов зарядки, поддерживаемых EVCC.

[V2G2-167] Каждая запись в списке поддерживаемых EVCC протоколов должна включать ProtocolNamespace, VersionNumberMajor и VersionNumberMinor, уникальный SchemaID, динамически назначенный EVCC, и приоритет (Priority) позиции протокола в списке. Приоритет в сообщении-запросе EVCC позволяет EVCC объявить предпочтительный протокол на уровне приложения, где Priority, равный 1, указывает на наивысший приоритет и Priority, равный 20, указывает на низший приоритет. Число протоколов, включенных в сообщение-запрос, ограничивается 20.

[V2G2-168] SECC должен отвечать сообщением supportedAppProtocolRes, как показано на рисунке 18, указав протокол, подлежащий использованию при последующем обмене сообщениями EVCC и SECC.

[V2G2-169] Сообщение-ответ должно включать ResponseCode и SchemaID протокола/схемы, согласованных в качестве прикладного протокола для последующего сеанса связи. Таким образом, SECC должен выбрать из собственного списка поддерживаемых протоколов протокол с наибольшим приоритетом, имеющийся в списке EVCC.

[V2G2-170] SECC должен подтвердить (положительно ответить на) поддерживаемый EVCC протокол, даже если значение VersionNumberMinor в сообщении-запросе EVCC не соответствует VersionNumberMinor поддерживаемого SECC протокола, в то время как VersionNumberMajor совпадает.

Примечание — Более высокое значение VersionNumberMinor указывает на то, что (по сравнению с меньшим значением) дополнительные элементы данных будут передаваться либо от EVCC, либо от SECC. Реализации, поддерживающие только более низкое значение VersionNumberMinor, могут быть не способны обрабатывать данные и могут быть вынуждены их игнорировать, однако разница значения VersionNumberMinor между EVCC и SECC не приводит к несовместимости. См. 8.2.4 с примерами успешного согласования протокола.

[V2G2-171] Все дополнительные элементы данных, которые определены соответствующей младшей версией, должны кодироваться как случай отклонения от схемы кодером EXI (см. также настройки опций EXI в 7.9.1.3).

[V2G2-172] Обычно ожидается, что SECC способен поддерживать соответствующие протоколы прикладного уровня, указанные EVCC. Однако когда ни один из протоколов прикладного уровня, включенных в список, полученный от EVCC, не поддерживается SECC, ResponseCode в сообщении-ответе должен быть равен Failed_NoNegotiation, указывая на то, что согласование протокола не было успешным. В этом сценарии ошибки сообщение с ответом не должно включать SchemaID.

[V2G2-173] Если успешное согласование протокола не может быть достигнуто, то EVCC не должен инициализировать сеанс связи.

[V2G2-174] Данное подтверждение протокола между EVCC и SECC должно выполняться до непосредственного обмена сообщениями на уровне приложения V2G. За исключением незначительных отклонений, в потоке сообщений V2G должен использоваться только набор сообщений, определенный в согласованном протоколе.

8.2.2 Определение сообщения supportedAppProtocolReq и supportedAppProtocolRes

[V2G2-175] EVCC и SECC должны реализовывать сообщение и элементы сообщения V2G, как показано на рисунке 17.

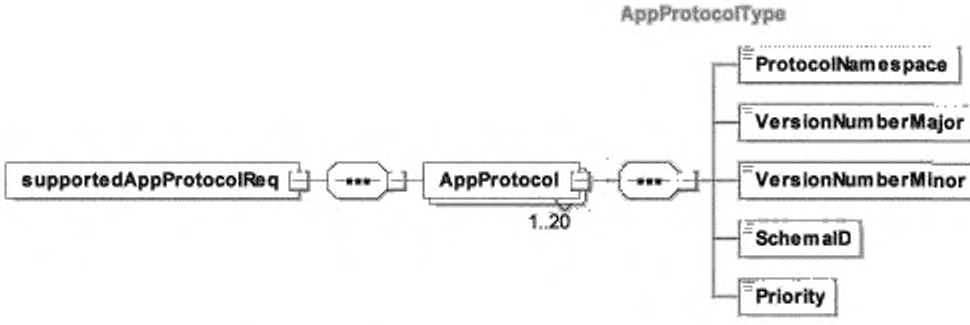


Рисунок 17 — Диаграмма — supportedAppProtocolReq

[V2G2-176] EVCC и SECC должны реализовывать сообщение и элементы сообщения V2G, как показано на рисунке 18.

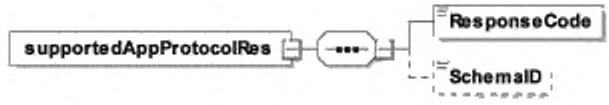


Рисунок 18 — Диаграмма — supportedAppProtocolRes

8.2.3 Описание семантики сообщений supportedAppProtocol

[V2G2-178] Элементы сообщений, показанные на рисунках 17 и 18, должны использоваться, как определено в таблицах 21 и 22.

Таблица 21 — Семантика и определение типа для элементов сообщения supportedAppProtocolReq

Имя элемента/признака	Тип	Семантика
AppProtocol	complexType: включает элементы сообщений, которые описаны в этой таблице	Этот элемент сообщения используется EVCC для передачи списка поддерживаемых протоколов. Каждый протокол с конкретной версией, поддерживаемый EVCC, представляется одной записью AppProtocol в сообщении-запросе (максимальное число записей: 20)
ProtocolNamespace	simpleType: protocolNamespaceType цепочка (макс. длина: 100) см. определение типа в F.2 (приложение F)	Этот элемент сообщения используется EVCC для уникальной идентификации Namespace URI конкретного протокола, поддерживаемого EVCC, т. е. это имя соответствующего протокола
VersionNumberMajor	simpleType: unsignedInt см. определение типа в F.2 (приложение F)	Этот элемент сообщения используется EVCC для указания старшего номера версии протокола, указанного в элементе сообщения ProtocolNamespace
VersionNumberMinor	simpleType: unsignedInt см. определение типа в F.2 (приложение F)	Этот элемент сообщения используется EVCC для указания младшего номера версии протокола, указанного в элементе сообщения ProtocolNamespace
SchemaID	simpleType: unsignedByte см. определение типа в F.2 (приложение F)	Этот элемент сообщения используется EVCC для указания SchemaID, присвоенного EVCC протоколу, указанному в элементах сообщения ProtocolNamespace, VersionNumberMajor и VersionNumberMinor

Окончание таблицы 21

Имя элемента/признака	Тип	Семантика
Priority	simpleType: priorityType unsignedByte (диапазон 1..20) см. определение типа F.2 (приложение F) type definition	Этот элемент сообщения используется EVCC для указания приоритета конкретного протокола, что позволяет SECC выбрать протокол на основе приоритетов. См. также **[V2G2-167]**

Таблица 22 — Семантика и определение типа для элементов сообщения supportedAppProtocolRes

Имя элемента/признака	Тип	Семантика
SchemaID	simpleType: unsignedByte см. определение типа в F.2 (приложение F)	Дополнительно: этот элемент сообщения используется SECC для ссылки на один из поддерживаемых EVCC протоколов, полученных в сообщении-запросе
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.2 (приложение F)	Этот элемент сообщения используется SECC для указания: включает ли список протоколов, полученный от EVCC, по крайней мере один протокол, соответствующий протоколам, поддерживаемым SECC

8.2.4 Примеры сообщений

8.2.4.1 Приоритизация протоколов

Примеры 1 и 2 сообщения V2G иллюстрируют обмен сообщениями supportedAppProtocol между EVCC и SECC. В сообщении-запросе EVCC посылает SECC список поддерживаемых протоколов на уровне приложения с обозначенным приоритетом (ИСО 15118:2:2013 версия 2.0, ИСО 15118:2:2010 версия 1.0), где первый протокол имеет наивысший приоритет. В ответном сообщении SECC подтверждает протокол ИСО 15118:2:2013 версия 2.0 с использованием ResponseCode, равного OK_SuccessfulNegotiation, и SchemaID, равного десяти (10).

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:supportedAppProtocolReq xsi:schemaLocation="urn:iso:15118:2:2010:AppProtocol ../V2G_CI_AppProtocol.xsd"
xmlns:n1="urn:iso:15118:2:2010:AppProtocol"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <AppProtocol>
        <ProtocolNamespace>urn:iso:15118:2:2013:MsgDef</ProtocolNamespace>
        <VersionNumberMajor>2</VersionNumberMajor>
        <VersionNumberMinor>0</VersionNumberMinor>
        <SchemaID>10</SchemaID>
        <Priority>1</Priority>
    </AppProtocol>
    <AppProtocol>
        <ProtocolNamespace>urn:iso:15118:2:2010:MsgDef</ProtocolNamespace>
        <VersionNumberMajor>1</VersionNumberMajor>
        <VersionNumberMinor>0</VersionNumberMinor>
        <SchemaID>20</SchemaID>
        <Priority>2</Priority>
    </AppProtocol>
</n1:supportedAppProtocolReq>
```

gost.e-ecolog.ru

Пример 1 сообщения V2G — supportedAppProtocolReq: приоритизация протоколов

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:supportedAppProtocolRes xsi:schemaLocation="urn:iso:15118:2:2010:AppProtocol ../V2G_CI_AppProtocol.xsd"
xmlns:n1="urn:iso:15118:2:2010:AppProtocol"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <ResponseCode>OK_SuccessfulNegotiation</ResponseCode>
    <SchemaID>10</SchemaID>
</n1:supportedAppProtocolRes>
```

Пример 2 сообщения V2G — supportedAppProtocolRes: приоритизация протоколов

8.2.4.2 Отклонение младшего номера версии

Примеры 3 и 4 сообщения V2G иллюстрируют обмен сообщениями supportedAppProtocol между EVCC и SECC. В сообщении-запросе EVCC посылает SECC только один поддерживаемый про-

токол прикладного уровня (ИСО 15118:2:2013 версия 2.0). SECC поддерживает только версию 2.1 протокола. В сообщении-ответе SECC подтверждает протокол ИСО15118:2:2013 посредством VersionNumberMajor, равного двум (2), используя SchemaID, равный одному (1). Однако ResponseCode равен OK_SuccessfulNegotiationWithMinorDeviation, что указывает на то, что имеет место отклонение младшего номера версии. EVCC может в таком случае ожидать элементов сообщений, которые неизвестны EVCC и могут быть проигнорированы.

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:supportedAppProtocolRes xsi:schemaLocation="urn:iso:15118:2:2010:AppProtocol  /V2G_CI_AppProtocol.xsd"
xmlns:n1="urn:iso:15118:2:2010:AppProtocol"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <AppProtocol>
        <ProtocolNamespace>urn:iso:15118:2:2013:MsgDef</ProtocolNamespace>
        <VersionNumberMajor>2</VersionNumberMajor>
        <VersionNumberMinor>0</VersionNumberMinor>
        <SchemaID>1</SchemaID>
        <Priority>1</Priority>
    </AppProtocol>
</n1:supportedAppProtocolReq>
```

Пример 3 сообщения V2G — supportedAppProtocolReq: отклонение младшего номера версии

```
<?xml version="1.0" encoding="UTF-8"?>
<n1:supportedAppProtocolRes xsi:schemaLocation="urn:iso:15118:2:2010:AppProtocol  /V2G_CI_AppProtocol.xsd"
xmlns:n1="urn:iso:15118:2:2010:AppProtocol"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <ResponseCode>OK_SuccessfulNegotiationWithMinorDeviation</ResponseCode>
    <SchemaID>1</SchemaID>
</n1:supportedAppProtocolRes>
```

Пример 4 сообщения V2G — supportedAppProtocolRes: отклонение младшего номера версии

8.3 Определение сообщения V2G

8.3.1 Обзор

В данном подразделе описываются сообщения V2G и их содержание. Он состоит из следующих трех пунктов:

- определение сообщения V2G (см. 8.3.2);
- определение заголовка сообщения V2G (см. 8.3.3);
- определение тела сообщения V2G (см. 8.3.4).

Примечание — Код XML-схемы приведен в приложении F.

На набор сообщений на уровне приложения указывает область имен XML-схемы «urn:iso:15118:2:2013:MsgDef». Подробности относительно определений областей дополнительных имен, используемых для определения сообщения, приведены в определении XML-схемы в приложении F.

8.3.2 Определение сообщения

На рисунке 19 показано определение схемы сообщения V2G прикладного уровня.

[V2G2-179] EVCC и SECC должны реализовывать структуру сообщения V2G, как показано на рисунке 19.

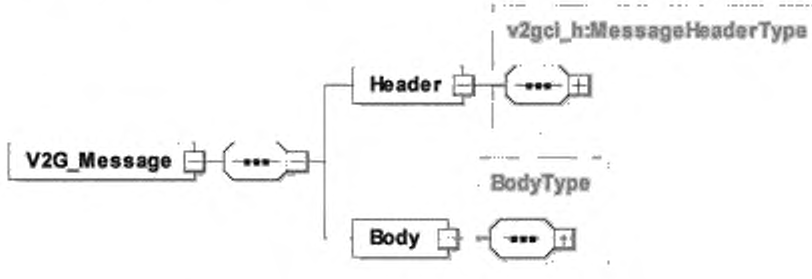


Рисунок 19 — Диаграмма — Сообщение V2G

[V2G2-180] Элементы данного сообщения должны использоваться, как определено в таблице 23.

Таблица 23 — Семантика и определение типа для сообщения V2G

Имя элемента	Тип	Семантика
Сообщение V2G	complexType включает элементы сообщений, которые описаны в данной таблице	Корневой элемент, который идентифицирует данный XML-документ как сообщение V2G. Он содержит два дочерних элемента: заголовок и тело
Заголовок	complexType: MessageHeaderType см. 8.3.3	Этот элемент заключает в себе содержание заголовка сообщения. Он включает общую информацию для потока протокола и не связан непосредственно с семантикой каждого конкретного сообщения, которое определено в пункте 0
Тело	complexType: BodyType см. 8.3.4	Этот элемент заключает в себе содержание тела сообщения. Тело сообщения дает фактическую семантику каждого сообщения, определенного в пункте 0

Пример 5 сообщения V2G показывает образец сообщения SessionSetupReq. Заголовок содержит SessionID, равный нулю (0), потому что новый сеанс связи должен вскоре начаться. Тело заключает в себе специфичное содержание сообщения. В этом случае сообщение содержит элемент сообщения EVCCID.

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns xmlsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns v2gci_t="urn iso 15118:2 2013:MsgDataTypes"
xmlns v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h.SessionID>00</v2gci_h SessionID>
    </v2gci_d:Header>
    <v2gci_d Body>
        <v2gci_b:SessionSetupReq>
            <v2gci_b:EVCCID>0123456789AB</v2gci_b EVCCID>
        </v2gci_b:SessionSetupReq>
    </v2gci_d Body>
</v2gci_d:V2G_Message>
```

Пример 5 сообщения V2G — Пример сообщения SessionSetupReq

8.3.3 Определение заголовка сообщения

Заголовок сообщения содержит общую информацию, которая включена во все сообщения. На рисунке 20 показано определение схемы заголовка сообщения V2G.

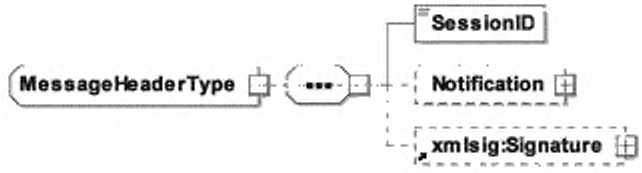


Рисунок 20 — Диаграмма — Заголовок сообщения

[V2G2-181] Элементы заголовка сообщения должны использоваться, как определено в таблице 24.

Таблица 24 — Семантика и определение типа для заголовка сообщения V2G

Имя элемента	Тип	Семантика
SessionID	simpleType: SessionIDType: hexBinary (макс. длина: 8)	Этот элемент сообщения используется EVCC и SECC для уникальной идентификации сеанса связи V2G. Требования, касающиеся данного элемента сообщений, см. в 8.4.2
Notification	complexType: NotificationType см. 8.5.2.8	Дополнительно: этот элемент используется SECC для передачи дополнительной информации об ошибке при ее возникновении на SECC
xmlsig:Signature	separate namespace: «gost.e-ecolog.ru 09/ xmldsig#»	Дополнительно: этот элемент используется, если требуется подписание определенного сообщения V2G

[V2G2-182] Каждое сообщение V2G, содержащее подписанные элементы, должно включать в заголовке элемент xmlsig:Signature, чтобы сделать возможной передачу подписи, сопровождающей элементы сообщения подписанного тела соответствующего сообщения.

8.3.4 Определение тела сообщения

Тело сообщения содержит данные, относящиеся к конкретному сообщению. На рисунке 21 показано определение схемы тела сообщения V2G. Сообщения, описанные ниже, являются производными от BodyBaseType, который представляет собой абстрактное содержание сообщения (см. 8.3.2). Различные прикладные сообщения определяются элементом BodyElement.



Рисунок 21 — Диаграмма — Тело сообщения

[V2G2-183] BodyElement должен использоваться, как определено в таблице 25.

Таблица 25 — Семантика и определение типа для тела сообщения V2G

Имя элемента	Тип	Семантика
BodyElement	complexType: BodyBaseType	BodyElement является начальным элементом группы подстановок и сам не входит в состав экземпляра сообщения. Вместо этого создается экземпляр одного из элементов тела

8.4 Определения сеанса связи V2G и элементов тела сообщения

8.4.1 Общие положения

Сеанс связи V2G в настоящем стандарте определяется как ассоциация двух конкретных субъектов V2G для обмена сообщениями V2G в предопределенной последовательности (см. 8.8) для управления процессом зарядки. Сеанс связи V2G всегда начинается с пары сообщений SessionSetupReq/Res и всегда заканчивается парой сообщений SessionStopReq/Res.

Все сообщения сеанса связи V2G несут SessionID, который позволяет управлять сеансами связи V2G между субъектами V2G на прикладном уровне. SessionID согласовывается EVCC и SECC в паре сообщений SessionSetupReq/Res. Все сообщения V2G сеанса связи V2G, за исключением сообщения SessionSetupReq, используют один и тот же SessionID.

SessionID позволяет приостанавливать и возобновлять сеанс зарядки с помощью нескольких сеансов связи V2G. Для этого EVCC и SECC применяют один и тот же SessionID во всех сеансах связи V2G во время сеанса зарядки. Применение нескольких сеансов связи V2G для приостановки и возобновления сеанса зарядки позволяет EVCC и SECC останавливать все уровни и перезапускать все уровни при возобновлении сеанса с сохранением всего прикладного контекста и данных управления процессом зарядки. Это, например, позволяет EV и EVSE полностью отключить модуль коммуникации во время паузы для экономии энергии.

Пауза сеанса связи V2G управляется параметром ChargingSession в SessionStopReq со значениями «Terminate» и «Pausing». Параметр может быть использован независимо от профиля зарядки. Это означает, что EV может инициировать паузу в любое время после отправки PowerDeliveryReq с ChargeProgress, равным «Stop» и в соответствии с **[V2G2-739]**.

На рисунке 22 показана обработка сеанса связи V2G.

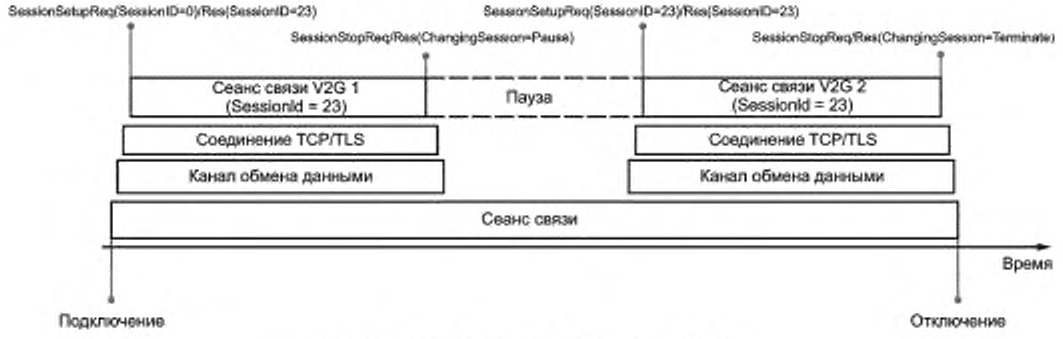


Рисунок 22 — Обработка сеанса связи V2G

8.4.2 Обработка сеанса

К EVCC применяются следующие требования:

[V2G2-739] EVCC должен сделать паузу в сеансе связи V2G с помощью параметра ChargingSession, установленного на значение «Pause» в сообщении SessionStopReq, и должен остановить сеанс связи V2G (прекратить связь на транспортном уровне).

Примечание 1 — При паузе сеанса связи V2G EVCC прекращает свою связь на транспортном уровне, что вызывает также паузу в процессе зарядки. Во время паузы зарядки также невозможно использование дополнительных услуг.

[V2G2-740] Если EVCC возобновляет ранее приостановленный сеанс связи V2G, следующие значения параметров, предоставленные EVCC в предыдущем сеансе связи V2G, должны быть предоставлены вновь для возобновляемого сеанса связи V2G:
- SessionID, который был передан в заголовке сообщения SessionSetupRes в предыдущем сеансе связи V2G (для всех сообщений-запросов, начиная с SessionSetupReq);
- SelectedPaymentOption (PaymentServiceSelectionReq);
- RequestedEnergyTransferMode (ChargeParameterDiscoveryReq).

[V2G2-742] Если EVCC желает возобновить ранее приостановленный сеанс связи V2G, он должен отправить параметр DepartureTime в ChargeParameterDiscoveryReq, уменьшенный на прошедшее время.

[V2G2-743] Если EVCC желает возобновить ранее приостановленный сеанс связи V2G, он должен отправить параметр EAmount в ChargeParameterDiscoveryReq, уменьшенный на уже отобранное количество энергии.

[V2G2-744] Если применяется **[V2G2-739]**, EVCC должен обеспечить выполнение **[V2G2-740]** при условии, что состояние A, E или F линии управления не было выявлено EVCC в соответствии с *ГОСТ Р МЭК 61851-1*.

[V2G2-746] При отправлении первого сообщения SessionSetupReq после подключения EV к EVSE EVCC должен установить параметр SessionID в заголовке сообщения, равный нулю (0).

[V2G2-747] Заголовок любого сообщения, отправленный EVCC во время активного сеанса связи V2G, должен включать значение SessionID, отправленное SECC в ответ на SessionSetupReq, инициирующий текущий активный сеанс связи V2G.

[V2G2-748] EVCC может возобновить сеанс зарядки путем отправления SesstionSetupReq с заголовком сообщения, в который входит значение SessionID из ранее приостановленного сеанса связи V2G.

[V2G2-749] Если сеанс связи V2G возобновляется в соответствии с **[V2G2-754]** и EVCC желает изменить параметры зарядки, перечисленные в **[V2G2-740]**, он должен использовать механизм пересогласования, описанный в настоящем стандарте.

К SECC применяются следующие требования:

[V2G2-741] Если EVCC возобновляет ранее приостановленный сеанс связи V2G, следующие значения параметров, предоставленные SECC в предыдущем сеансе связи V2G, должны быть предоставлены вновь для возобновляемого сеанса связи V2G:

- SessionID, который был передан в заголовке сообщения SessionSetupRes в предыдущем сеансе связи V2G, если SessionID, переданный в заголовке SessionSetupReq, соответствует сохраненному значению (для всех сообщений с запросом, начиная с SessionSetupReq);
- PaymentOptionList (ServiceDiscoveryRes). Должна предоставляться только опция оплаты, ранее выбранная EVCC;
- ChargeService (ServiceDiscoveryRes);
- SAScheduleTuple (ChargeParameterDiscoveryRes). Должен предоставляться по крайней мере SAScheduleTuple (включая соответствующие данные PMaxSchedule и SalesTariff), идентификатор которого был выбран EVCC в его ChargingProfile в предыдущем сеансе связи V2G. Этот идентификатор кортежа не должен меняться во время сеанса связи V2G. Период времени, в течение которого данный SAScheduleTuple применяется, должен быть уменьшен на уже прошедшее время.

Примечание 2 — Необходимо, чтобы ChargingProfile, рассчитанный EVCC во время предыдущего(их) сеанса(ов) связи V2G, оставался действительным после прерывания зарядки для всего сеанса зарядки.

Примечание 3 — EV может делать выбор между следованием сохраненному ChargingProfile или созданием нового ChargingProfile на базе дополнительных SAScheduleTuples, которые даны в SAScheduleList сообщения ChargeParameterDiscoveryRes.

[V2G2-745] Если **[V2G2-739]** применяется, SECC должен обеспечить выполнение **[V2G2-741]** при условии, что состояние A, E или F не было выявлено EVCC в соответствии *с ГОСТ Р МЭК 61851-1*.

[V2G2-750] При получении SessionSetupReq с параметром SessionID, равным нулю (0), SECC должен генерировать новое (несохраненное) значение SessionID, отличающееся от нуля (0), и отправить это значение в заголовке сообщения SessionSetupRes.

[V2G2-751] Значение SessionID, отправленное SECC в сообщении SessionSetupRes, не должно изменяться до прекращения сеанса связи V2G (т. е. до установки параметра ChargingSession сообщения SessionStopReq в какой-либо момент времени в значение «Terminate»).

[V2G2-752] Заголовок любого сообщения, отправленный SECC во время активного сеанса связи V2G, должен включать значение SessionID, отправленное SECC в ответ на SessionSetupReq, инициирующий текущий активный сеанс связи V2G.

[V2G2-753] Если EVCC решает возобновить сеанс зарядки, отправив SesstionSetupReq с заголовком сообщения, в который входит значение SessionID приостановленного ранее сеанса связи V2G, SECC должен сравнить это значение со значением, сохраненным в предшествующем сеансе связи V2G.

[V2G2-754] Если значение SessionID, полученное в текущем SessionSetupReq, равно значению, сохраненному из предшествующего сеанса связи V2G, SECC должен подтвердить продолжение сеанса зарядки путем отправки сообщения SessionSetupRes, включив сохраненное значение SessionID и указав на возобновление сеанса связи V2G посредством ResponseCode, установленного в значение «OK_OldSessionJoined» (см. также **[V2G2-463]** относительно выбора надлежащего кода ответа).

[V2G2-755] Если сеанс связи V2G возобновляется в соответствии с **[V2G2-754]** и SECC желает изменить параметры зарядки, перечисленные в **[V2G2-741]**, он должен использовать механизм пересогласования, описанный в настоящем стандарте.

[V2G2-756] Если SECC получает SessionSetupReq, включающее значение SessionID, которое не равно нулю (0) и не равно значению SessionID, сохраненному в предшествующем сеансе связи V2G, он должен отправить в сообщении SessionSetupRes значение SessionID, которое не равно «0» и не равно значению SessionID, сохраненному в предшествующем сеансе связи V2G, и указать на новый сеанс связи V2G посредством ResponseCode, установленного в значение «OK_NewSessionEstablishe» (см. также **[V2G2-462]** относительно применимости этого кода ответа).

Примечание 4 — Дополнительные требования относительно использования описанных в настоящем стандарте кодов ответов, которые применимы к параметру SessionID, см. в 8.8.3.

8.4.3 Общие сообщения

8.4.3.1 Обзор

Сообщения, описанные как общие сообщения, могут быть применены в последовательности сообщений в любом режиме зарядки.

8.4.3.2 SessionSetupReq/Res

8.4.3.2.1 SessionSetupReq

С помощью сообщения SessionSetupReq EVCC устанавливает сеанс связи V2G.

[V2G2-188] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 23.

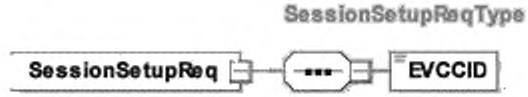


Рисунок 23 — Диаграмма — SessionSetupReq

[V2G2-189] Элементы данного сообщения должны использоваться, как определено в таблице 26.

Таблица 26 — Семантика и определение типа для SessionSetupReq

Имя элемента	Тип	Семантика
EVCCID	simpleType: EVCCIDType hexBinary (макс. длина: 6) см. определение типа в F.6 (приложение F)	Дает идентификатор EV в читаемом формате. Содержит MAC-адрес EVCC как шесть байтов в шестнадцатеричной кодировке, т. е. элемент должен иметь длину шесть байтов

[V2G2-879] EVCC должен передать EVCCID длиной шесть байтов и должен заполнить его своим MAC-адресом.

8.4.3.2.2 SessionSetupRes

SECC отвечает на SessionSetupReq сообщением SessionSetupRes. Посредством ResponseCode, включенного в сообщение SessionSetupRes, SECC сообщает EVCC, успешным или нет было установление нового сеанса или присоединение к предыдущему сеансу связи.

[V2G2-190] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 24.

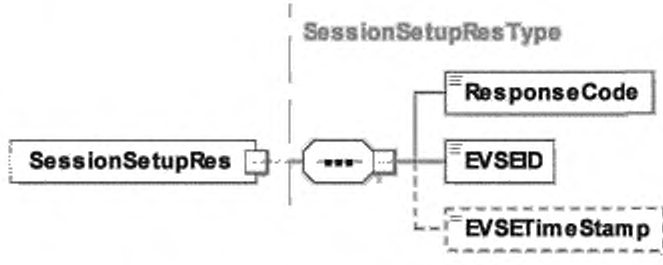


Рисунок 24 — Диаграмма — SessionSetupRes

[V2G2-191] Элементы данного сообщения должны использоваться, как определено в таблице 27.

Таблица 27 — Семантика и определение типа для SessionSetupRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
EVSEID	simpleType: evseIDType строка (мин. длина: 7, макс. длина: 37)	Любой идентификатор, уникальным образом идентифицирующий EVSE и точку отбора мощности, к которым подключено транспортное средство. Формат этого элемента сообщения определен в приложении G. Если SECC не может предоставить такие идентификационные данные, значение EVSEID устанавливается на ноль («ZZ00000»)
EVSETimeStamp	simpleType: long см. определение типа в F.6 (приложение F)	Дополнительно: метка текущего времени SECC. Формат «Метка времени Unix». Этот элемент сообщения может быть использован EVCC для решения о возможности использоования конкретного сертификата контракта для зарядки во время текущего сеанса связи. На основе этой информации EVCC может реализовать стратегию для обновления сертификатов. На основе этой информации EV может синхронизировать свое внутреннее время, если другие средства на EV отсутствуют

[V2G2-192] EVCC и SECC должны использовать формат для идентификатора EVSE (EVSEID), описанный в приложении G.

8.4.3.3 ServiceDiscoveryReq/Res

8.4.3.3.1 Обработка ServiceDiscoveryReq/Res

Служба обнаружения сервисов (Service Discovery) позволяет EVCC найти все услуги, предоставляемые SECC. В настоящем стандарте описаны свойства интерфейса между EVCC и SECC, относящиеся к зарядке EV. Служба обнаружения сервисов различает типы и объемы услуг.

8.4.3.3.2 ServiceDiscoveryReq

Отправив сообщение ServiceDiscoveryReq, EVCC побуждает SECC к отправке информации о всех услугах, предлагаемых SECC. Кроме того, EVCC может ввести ограничения для определенных сервисов с помощью элементов объема сервиса и типа сервиса.

[V2G2-193] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 25.

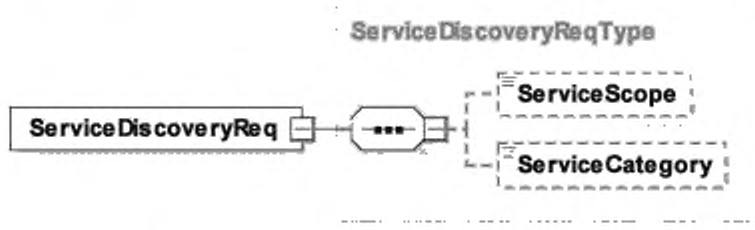


Рисунок 25 — Диаграмма — ServiceDiscoveryReq

[V2G2-194] Элементы данного сообщения должны использоваться, как определено в таблице 28.

Таблица 28 — Семантика и определение типа для ServiceDiscoveryReq

Имя элемента	Тип	Семантика
ServiceScope	simpleType: serviceScopeType строка (макс. длина: 32) см. определение типа в F.6 (приложение F)	Дополнительно: определяет диапазон обнаружения сервисов. Диапазон определяется уникальным унифицированным идентификатором ресурса (URI), который соответствует поставщику услуг (например, поставщику услуг для EV, поставщику дополнительных услуг и т. п.). С применением диапазона к обнаружению сервисов итоговый перечень услуг, передаваемый в ServiceDiscoveryRes, может быть ограничен определенным объемом, что обеспечивает предварительный отбор. SECC всегда передает все поддерживаемые сервисы всех видов, если никакой конкретный объем услуг ServiceScope не был указан в сообщении-запросе. Определение URI (унифицированный идентификатор ресурса) описано в *[48]* и зависит от специфики реализации
ServiceCategory	simpleType: serviceCategoryType enumeration см. определение типа в F.6 (приложение F)	Дополнительно: определяет категорию услуги для обнаружения сервиса (например, зарядка EV, доступ к Интернету и т. д.). С применением категории к обнаружению сервиса итоговый перечень услуг, передаваемый в ServiceDiscoveryRes, может быть ограничен определенной категорией сервисов, что обеспечивает предварительный отбор. SECC всегда передает все поддерживаемые сервисы для всех категорий, если никакая конкретная категория не была указана в сообщении-запросе с помощью элемента ServiceCategory сообщения

8.4.3.3.3 ServiceDiscoveryRes

После получения сообщения ServiceDisoveryReq от EVCC SECC посылает сообщение ServiceDiscoveryRes. В случае успешного обнаружения сервиса в ответе перечисляются все имеющиеся услуги SECC для обозначенных критериев. В случае необнаружения сервиса список сервисов пустой и ResponseCode указывает потенциальные причины этого.

[V2G2-195] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 26.

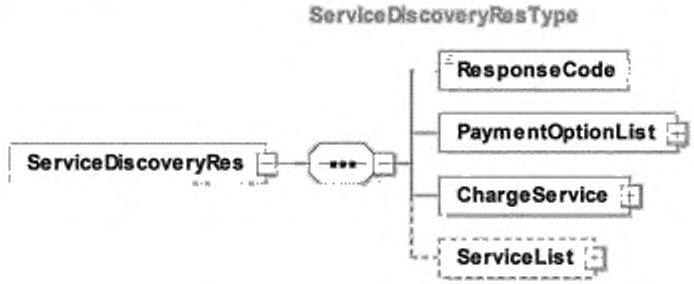


Рисунок 26 — Диаграмма — ServiceDiscoveryRes

[V2G2-196] Элементы данного сообщения должны использоваться, как определено в таблице 29.

Таблица 29 — Семантика и определение типа для ServiceDiscoveryRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
PaymentOptionList	complexType: PaymentOptionListType см. 8.5.2.9	Этот элемент включает список опций оплаты, который SECC предлагает EVCC, указывая методы, которые могут быть выбраны для оплаты услуг. EVCC может выбрать только один метод оплаты для всех услуг, используемых им
ChargeService	complexType: ChargeServiceType см. 8.5.2.3	Имеющиеся услуги зарядки, которые поддерживает EVSE
ServiceList	complexType: ServiceListType см. 8.5.2.2	Факультативно: список, содержащий информацию о всех других услугах, кроме зарядных, которые предлагает EVSE. Возвращаемый список услуг является перечнем услуг, выбранных на основе ServiceScope и ServiceType, указанных в сообщении ServiceDiscoveryReq. Число сервисных элементов ограничивается восемью

8.4.3.4 ServiceDetailReq/Res

8.4.3.4.1 ServiceDetailReq

Отправив сообщение ServiceDetailReq, EVCC запрашивает SECC об отправке конкретной дополнительной информации о сервисах, предлагаемых EVSE.

[V2G2-197] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 27.



Рисунок 27 — Диаграмма — ServiceDetailReq

[V2G2-198] Элементы данного сообщения должны использоваться, как определено в таблице 30.

Таблица 30 — Семантика и определение типа для ServiceDetailReq

Имя элемента	Тип	Семантика
ServiceID	simpleType: serviceIDType unsignedShort см. определение типа в F.6 (приложение F)	Этот элемент идентифицирует сервис, который был предложен SECC в сообщении ServiceDiscoveryRes

8.4.3.4.2 ServiceDetailRes

После получения сообщения ServiceDetailReq от EVCC SECC посылает сообщение ServiceDetailRes и предоставляет подробности о сервисах.

[V2G2-199] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 28.

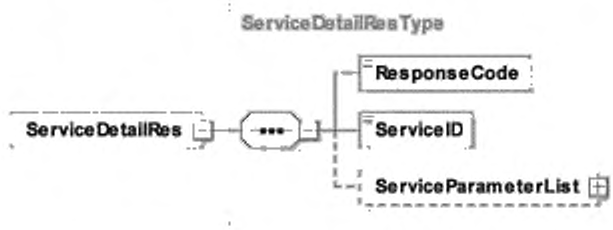


Рисунок 28 — Диаграмма — ServiceDetailRes

[V2G2-200] Элементы данного сообщения должны использоваться, как определено в таблице 31.

Таблица 31 — Семантика и определение типа для ServiceDetailRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
ServiceID	simpleType: serviceIDType unsignedShort см. определение типа в F.6 (приложение F)	Этот элемент идентифицирует сервис, который был предложен SECC в сообщении ServiceDiscoveryRes
ServiceParameterList	complexType: ServiceParameterListType см. 8.5.2.21	Включает список параметров для конкретного serviceID, полученный от SECC в сообщении ServiceDiscoveryRes

8.4.3.5 PaymentServiceSelectionReq/Res

8.4.3.5.1 Обработка выбора оплаты и сервиса

На основе услуг и соответствующих опций оплаты, предоставляемых SECC, эта пара сообщений позволяет передавать выбранные опции платежа (PaymentOption), выбранные услуги (SelectedServices) и соответствующие наборы параметров (ParameterSets). В зависимости от выбранного типа оплаты происходит обмен дополнительными сообщениями (пара сообщений PaymentDetails).

8.4.3.5.2 PaymentServiceSelectionReq

Это сообщение-запрос передает информацию о выбранных услугах и о том, как все выбранные услуги оплачиваются (см. 8.6.3.1).

[V2G2-201] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 29.

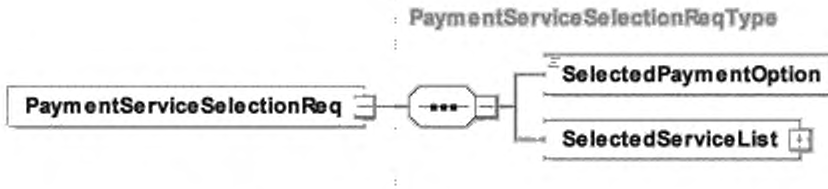


Рисунок 29 — Диаграмма — PaymentServiceSelectionReq

[V2G2-202] Элементы данного сообщения должны использоваться, как определено в таблице 32.

Таблица 32 — Семантика и определение типа для PaymentServiceSelectionReq

Имя элемента	Тип	Семантика
SelectedPaymentOption	simpleType: paymentOptionType enumeration см. определение типа в F.6 (приложение F)	Этот элемент используется для указания выбранного типа оплаты за использование всех выбранных сервисов в selectedServiceList
SelectedServiceList	complexType: SelectedServiceListType см. 8.5.2.24	Список содержит идентификаторы всех выбранных сервисов (ServiceIDs) и факультативный parameterSetID для соответствующего serviceID

8.4.3.5.3 PaymentServiceSelectionRes

Этим сообщением SECC информирует EVCC, были ли приняты выбранные услуги и опция оплаты. В зависимости от выбранного типа оплаты происходит обмен дополнительными сообщениями (пара сообщений PaymentDetails).

[V2G2-203] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 30.



Рисунок 30 — Диаграмма — PaymentServiceSelectionRes

[V2G2-204] Элементы данного сообщения должны использоваться, как определено в таблице 33.

Таблица 33 — Семантика и определение типа для PaymentServiceSelectionRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC

8.4.3.6 PaymentDetailsReq/Res

8.4.3.6.1 Обработка ServiceDiscoveryReq/Res

Структура сообщения о платежных реквизитах используется, только когда необходим обмен определенными платежными реквизитами (например, в случае зарядки на основе контракта будет необходим eMAID).

8.4.3.6.2 PaymentDetailsReq

Посредством PaymentDetailsReq EVCC предоставляет платежные реквизиты, если выбранный способ оплаты был «Contract». EVCC запрашивает у SECC отправку вызова, используя сертификат подписи и eMAID.

[V2G2-205] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 31.

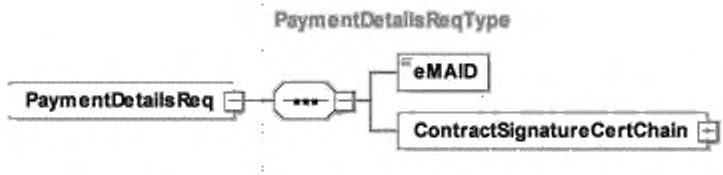


Рисунок 31 — Диаграмма — PaymentDetailsReq

[V2G2-206] Элементы данного сообщения должны использоваться, как определено в таблице 34.

Таблица 34 — Семантика и определение типа для PaymentDetailsReq

Имя элемента	Тип	Семантика
eMAID	simpleType: eMAIDType строка (мин. длина: 14, макс. длина: 15) см. определение типа в F.6 (приложение F)	Этот элемент идентифицирует контракт на зарядку. Формат определен в приложении G
ContractSignatureCert Chain	complexType: CertificateChainType см. 8.5.2.4	Этот элемент содержит сертификат контракта на зарядку и факультативные субсертификаты

8.4.3.6.3 PaymentDetailsRes

С помощью сообщения PaymentDetailsRes SECC информирует EVCC, были ли одобрены платежные реквизиты. Кроме того, SECC направляет запрос в виде случайного числа, который должен быть подписан EVCC.

[V2G2-208] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 32.

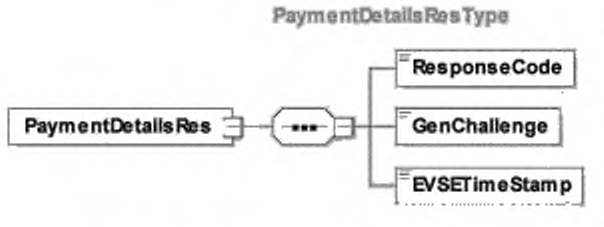


Рисунок 32 — Диаграмма — PaymentDetailsRes

[V2G2-209] Элементы данного сообщения должны использоваться, как определено в таблице 35.

Таблица 35 — Семантика и определение типа для PaymentDetailsRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения

Окончание таблицы 35

Имя элемента	Тип	Семантика
GenChallenge	simpleType: genChallengeType base64Binary (длина 16) см. определение типа в F.6 (приложение F)	Вызов, отправленный SECC. Этот элемент содержит генерированное произвольное число
EVSETimeStamp	simpleType: long см. определение типа в F.6 (приложение F)	Метка текущего времени SECC. Формат «Метка времени Unix». Этот элемент сообщения может быть использован EVCC для определения причины отклонения сертификата контракта со стороны SECC и/или цепочки сигналов, переданных от EVCC к SECC в сообщении «PaymentDetailsReq». На основе этой информации EVCC может реализовать стратегию обновления сертификатов (для будущих сеансов). На основе этой информации EV может синхронизировать свое внутреннее время, если другие средства отсутствуют

[V2G2-825] Поле GenChallenge должно быть длиной 128 бит.

[V2G2-826] Энтропия поля GenChallenge должна быть не менее 120 бит.

[V2G2-898] В случае выполнения идентификационного режима PnC EVCC должен отправить свой сертификат контракта, включая цепочку сертификата (до корневого сертификата, но исключая его) в элементе ContractSignatureCertChain сообщения PaymentDetailsReq.

[V2G2-899] В случае выполнения идентификационного режима PnC SECC должен получить сертификат контракта, включая цепочку сертификата, в элементе ContractSignatureCertChain сообщения PaymentDetailsReq. SECC должен выполнить валидацию сертификата, включая цепочку сертификата, проверить, прослеживается ли он до заслуживающего доверия корневого сертификата оператора услуг. Если какая-либо из вышеуказанных проверок и валидаций не удается, SECC должен считать сертификат контракта недействительным. SECC также не должен использовать указанный сертификат контракта для каких-либо дальнейших проверок.

8.4.3.7 AuthorizationReq/Res

8.4.3.7.1 AuthorizationReq

Если сгенерированный вызов был послан SECC в предыдущем сообщении, EVCC транслирует данный вызов обратно с соответствующей подписью. В противном случае сообщение остается пустым.

[V2G2-210] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице104 и в соответствии с рисунком 33.

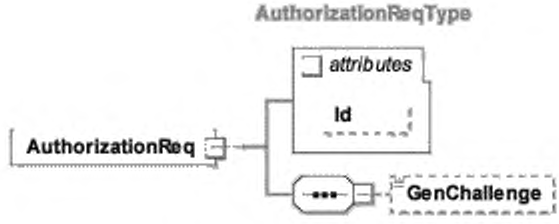


Рисунок 33 — Диаграмма — AuthorizationReq

[V2G2-211] Элементы данного сообщения должны использоваться, как определено в таблице 36.

Таблица 36 — Семантика и определение типа для AuthorizationReq

Имя элемента	Тип	Семантика
Id	simpleType: ID цепочка см. определение типа в F.6 (приложение F)	Факультативно: этот атрибут используется для ссылки на тело сообщения в заголовке подписи, если требуется применить подпись
GenChallenge	simpleType: genChallengeType base64Binary (длина 16) см. определение типа в F.6 (приложение F)	Факультативно: вызов, отправленный SECC в сообщении PaymentDetailsRes. Этот элемент содержит сгенерированное произвольное число

[V2G2-697] Поле GenChallenge должно быть длиной 128 бит.

[V2G2-698] Энтропия поля GenChallenge должна быть не менее 120 бит.

8.4.3.7.2 AuthorizationRes

В заключение SECC проверяет подпись вызова (и цепочку сертификата, если это не сделано раньше) и посылает соответствующее сообщение-ответ с авторизацией.

[V2G2-212] В зависимости от выбранного набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 34.

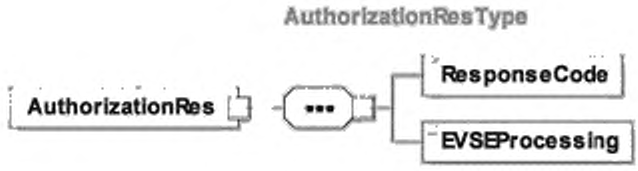


Рисунок 34 — Диаграмма — AuthorizationRes

[V2G2-213] Элементы данного сообщения должны использоваться, как определено в таблице 37.

Таблица 37 — Семантика и определение типа для AuthorizationRes

Имя элемента	Тип	Семантика
EVSEProcessing	simpleType: EVSEProcessingType enumeration см. определение типа в F.6 (приложение F)	Параметр, указывающий, что EVSE завершил обработку, которая была инициирована после AuthorizationReq, или, если EVSE в данное время продолжает обработку, что сообщение-ответ было отправлено
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	Параметр, показывающий статус подтверждения любого из сообщений V2G, полученных SECC

[V2G2-900] При выполнении идентификационного режима PnC, в случае если EVCC отправляет вызов (ранее полученный от SECC), EVCC должен подписать элемент тела AuthorizationReq, используя закрытый ключ сертификата контракта, который он отправил в PaymentDetailsReq в течение данного сеанса.

[V2G2-901] В случае выполнения идентификационного режима PnC SECC должен удостовериться в том, что элемент тела AuthorizationReq правильно подписан закрытым ключом, соответствующим сертификату контракта, ранее полученному в сообщении PaymentDetailsReq

данного сеанса, и что поле GenChallenge имеет такое же содержание, как поле GenChallenge, ранее отправленное SECC в сообщении PaymentsDetailsRes. Если какая-либо из указанных проверок и валидаций не удается, SECC должен считать подпись недействительной. См. также **[V2G2-475]**.

8.4.3.8 ChargeParameterDiscoveryReq/Res

8.4.3.8.1 Обработка ChargeParameterDiscoveryReq/Res

После авторизации зарядки на EVSE (SECC) EVCC и SECC согласовывают параметры зарядки с помощью пары сообщений ChargeParameterDiscoveryReq/Res.

Примечание — Последовательность сообщений для пересогласования описана в приложении Н.

Согласование параметров зарядки, которое предшествует передаче электроэнергии или во время фазы передачи, призвано обеспечить удовлетворение потребностей потребителя и одновременно эффективную передачу электроэнергии в соответствии с возможностями электросети на локальном (частная сеть) и региональном уровнях (сеть общего пользования). Согласование становится все более необходимым по мере увеличения числа EV, а также местного производства электроэнергии из возобновляемых источников.

До начала передачи электроэнергии EV проводит необходимые согласования с EVSE и опосредованно с третьими субъектами по обеспечению соответствия с текущими или прогнозируемыми объемами электроэнергии. Данные по наличию электроэнергии могут корректироваться после начала зарядки вследствие возникновения нехватки электроэнергии или увеличения расхода другими потребителями (например, другими EV). При этом должно выполняться новое согласование для решения задач по поставке электроэнергии в локальном или региональном масштабе.

Такое пересогласование должно включаться в протокол связи между EVSE и EV в период зарядки. Системы верхнего уровня, которые будут разрабатываться в будущем на основе данного протокола, будут сочетать указанные функции для достижения оптимального баланса между удовлетворением потребностей конечного потребителя и другими ограничениями.

8.4.3.8.2 ChargeParameterDiscoveryReq

При отправлении сообщения ChargeParameterDiscoveryReq EVCC передает параметры зарядки SECC. Данное сообщение содержит информацию о статусе EV и дополнительных параметрах зарядки, таких как примерное количество энергии для подзарядки EV, возможности системы для зарядки EV, время, когда водитель намерен уехать со станции EVSE.

[V2G2-214] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 35.

[V2G2-215] При отсутствии информации DepartureTime EVCC не должен включать этот элемент сообщения в сообщение ChargeParameterDiscoveryReq.

[V2G2-761] Если EVCC не посылает время отбытия в составе сообщения ChargeParameterDiscoveryReq, SECC должен исходить из того, что EV должен начать зарядку без какой-либо задержки.

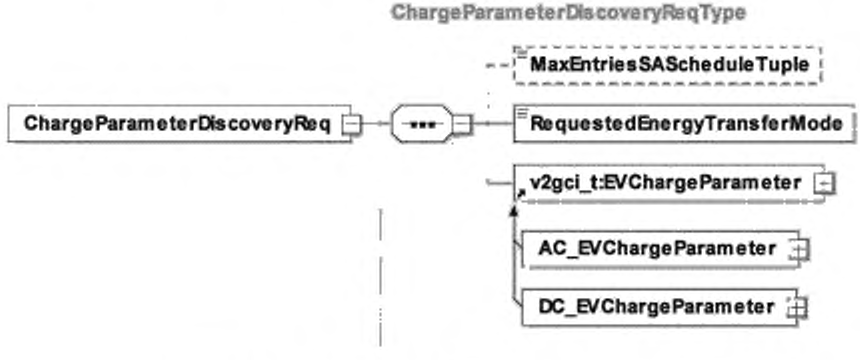


Рисунок 35 — Диаграмма — ChargeParameterDiscoveryReq

[V2G2-216] Элементы данного сообщения должны использоваться, как определено в таблице 38.

Таблица 38 — Семантика и определение типа для ChargeParameterDiscoveryReq

Имя элемента	Тип	Семантика
MaxEntriesSAScheduleTuple	simpleType: unsignedShort см. определение типа в F.6 (приложение F)	Факультативно: указывает максимальное число записей в SAScheduleTuple (применятся к Pmax и Tariff). EVSE может передавать число записей до максимального числа, которое определено в параметре
RequestedEnergyTransferMode	simpleType: EnergyTransferModeType enumeration см. определение типа в F.6 (приложение F) и таблице 63	Выбранный режим передачи энергии для зарядки, который запрашивается SECC
AC_EVChargeParameter	complexType: AC_EVChargeParameterType заменяет абстрактный тип EV_ChargeParameterType см. 8.5.3.2	Этот элемент используется EVCC для инициирования процесса назначения целей для зарядки переменным током
DC_EVChargeParameter	complexType: DC_EVChargeParameterType заменяет абстрактный тип EV_ChargeParameterType см. 8.5.4.3	Этот элемент используется EVCC для инициирования процесса назначения целей для зарядки постоянным током

Определение EnergyTransferModeType поддерживает соединители типа 1, 2 и 3 в соответствии с *ГОСТ Р МЭК 62196-2* и соединители в соответствии с *[49]* (приложения cc, dd, ee и ff). С учетом поддерживаемых соединителей EVCC может выбирать услуги по зарядке, как указано в 8.5.2.4.

[V2G2-784] EVCC должен поддерживать 12 записей для элементов PMaxScheduleEntry и SalesTariffEntry внутри одного SAScheduleTuple, если MaxEntriesSAScheduleTuple не передается в сообщении ChargeParameterDiscoveryReq.

[V2G2-786] SECC должен поддерживать минимум 12 записей для элементов PMaxScheduleEntry и SalesTariffEntry внутри одного SAScheduleTuple.

[V2G2-785] Если SECC не может обеспечить число записей для элементов PMaxScheduleEntry и SalesTariffEntry внутри одного SAScheduleTuple, которое требуется для MAXEntriesSAScheduleTuple, он должен отправить максимальное число записей, которое поддерживается в SAScheduleTuple.

8.4.3.8.3 ChargeParameterDiscoveryRes

Посредством сообщения ChargeParameterDiscoveryRes SECC передает параметры зарядки, которые применимы для электросети. В дополнение к общим параметрам зарядки для EVSE это сообщение может включать дополнительную информацию о стоимости электроэнергии в зависимости от времени, запроса, потребления или сочетания этих параметров. Термин «стоимость» относится к любого рода стоимости (см. 8.5.2.20), указанной в настоящем стандарте, и не ограничивается стоимостью в денежном выражении. На основе указанной информации о стоимости EV может оптимизировать график своей зарядки по запрашиваемому количеству энергии.

Примечание 1 — Посредством «стоимости» EVSE может стимулировать (не принуждать) EV к зарядке в периоды, которые благоприятны для электросети (например, при излишках получения ветровой энергии и т. п.).

[V2G2-218] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 36.

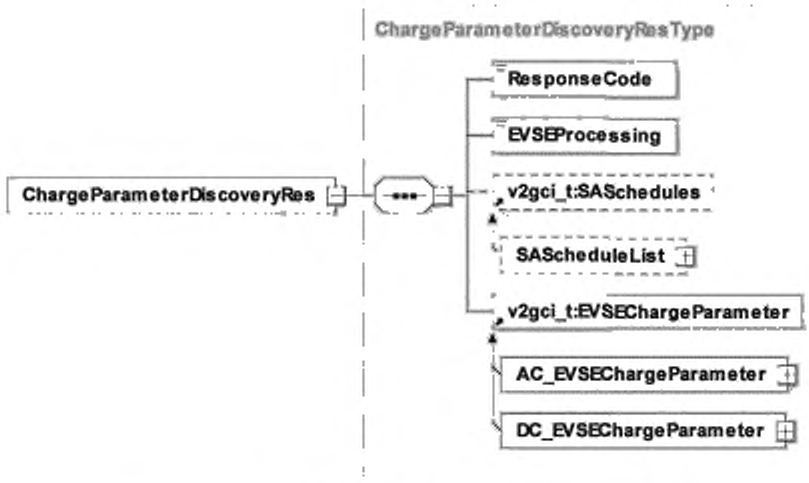


Рисунок 36 — Диаграмма — ChargeParameterDiscoveryRes

[V2G2-219] Все тарифы, представленные в элементе SASchedules, должны исходить от одного и того же SA.

[V2G2-220] Элементы данного сообщения должны использоваться, как определено в таблице 39.

Таблица 39 — Семантика и определение типа для ChargeParameterDiscoveryRes

Имя элемента	Тип	Семантика
EVSEProcessing	simpleType: EVSEProcessingType enumeration см. определение типа в F.6 (приложение F)	Параметр, показывающий, что EVSE завершил обработку, которая была инициирована после ChargeParameterDiscoveryReq, или что в это время EVSE продолжает обработку
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
SAScheduleList	complexType: SAScheduleListType заменяет абстрактный тип SASchedulesType см. 8.5.2.12	Факультативно: включает несколько графиков от SA. SECC не должен применять параметр «SAScheduleList», если EVSEProcessing установлен в значение «Ongoing»
AC_EVSECharge Parameter	complexType: AC_EVSEChargeParameterType заменяет абстрактный тип EVSE_ChargeParameterType см. 8.5.3.3	Этот элемент используется EVCC для инициирования процесса назначения целей для зарядки переменным током
DC_EVSECharge Parameter	complexType: DC_EVSEChargeParameterType заменяет абстрактный тип EVSE_ChargeParameterType см. 8.5.4.4	Этот элемент используется EVCC для инициирования процесса назначения целей для зарядки постоянным током

Примечание 2 — С помощью параметра EVSEProcessing EVSE может указать EVCC на то, что обработка информации не завершилась, но сообщение с ответом должно быть отправлено для выполнения требований по тайм-ауту и производительности, изложенных в 8.7.2. Это позволяет продолжить сеанс связи, выполняя требования по производительности и тайм-ауту.

8.4.3.9 PowerDeliveryReq/Res

8.4.3.9.1 Обработка PowerDeliveryReq/Res

Обмен сообщениями о начале зарядки указывает на момент времени, когда EVSE подает напряжение на зарядный разъем.

[V2G2-286] Элемент SAScheduleTupleID должен идентифицировать выбранный элемент SAScheduleTupleType (см. 8.5.2.13) в списке элементов SAScheduleTuple (см. 8.5.2.12), переданном в сообщении ChargeParameterDiscoveryRes (см. 8.4.3.8.3).

8.4.3.9.2 PowerDeliveryReq

С помощью сообщения PowerDeliveryReq EVCC запрашивает у SECC передачу электроэнергии и передает ChargingProfile, которому EVCC будет следовать во время зарядки.

Примечание 1 — Момент отправления этого сообщения не обязательно соответствует началу процесса зарядки. EV может корректировать параметры графика зарядки, когда процесс зарядки уже начался.

[V2G2-221] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны формировать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 37.

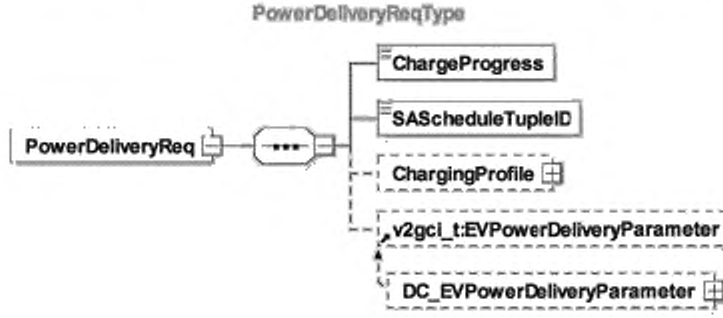


Рисунок 37 — Диаграмма — PowerDeliveryReq

[V2G2-222] Элементы данного сообщения должны использоваться, как определено в таблице 40.

Таблица 40 — Семантика и определение типа для PowerDeliveryReq

Имя элемента	Тип	Семантика
ChargeProgress	simpleType: chargeProgressType enumeration см. определение типа в F.6 (приложение F)	Этот элемент сообщения используется для запроса EVSE о выполнении условий передачи электроэнергии после того, как бортовая система EV начнет получать электроэнергию (т. е. у EVSE запрашивается замыкание контактов). Если ChargeProgress равен «Start», EVSE запрашивается о готовности к незамедлительному обеспечению электроэнергией. Если ChargeProgress равен «Stop», EVSE запрашивается о прекращении передачи электроэнергии. Если ChargeProgress равен «Renegotiate», энергия не передается и применяется механизм пересогласования, описанный в настоящем стандарте
SAScheduleTupleID	simpleType: SAIDType short см. определение типа в F.6 (приложение F)	Уникальный идентификатор сеанса зарядки для элемента SAScheduleTuple. Идентификатор SA остается уникальным идентификатором для одного графика в течение сеанса зарядки

Окончание таблицы 40

Имя элемента	Тип	Семантика
ChargingProfile	complexType: ChargingProfileType см. 8.5.2.10	Факультативно: позволяет EV резервировать конкретный профиль зарядки для текущего сеанса (т. е. максимальное количество получаемой энергии)
DC_EVPowerDelivery Parameter	complexType: DC_EVPowerDeliveryParameterType заменяет абстрактный тип EVPowerDeliveryParameter см. 8.5.4.5	Факультативно: этот элемент используется EVCC для передачи параметров мощности

Примечание 2 — В *ГОСТ Р 58122 (ИСО 15118-1:2013)* термин «ChargingProfile» («профиль зарядки») также называется «графиком зарядки».

8.4.3.9.3 PowerDeliveryRes

SECC после получения сообщения PowerDeliveryReq EVCC направляет сообщение PowerDeliveryRes, включая информацию об объеме электроэнергии.

[V2G2-223] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны генерировать обязательные сообщения и элементы сообщений, как описано в таблице 104 и в соответствии с рисунком 38.

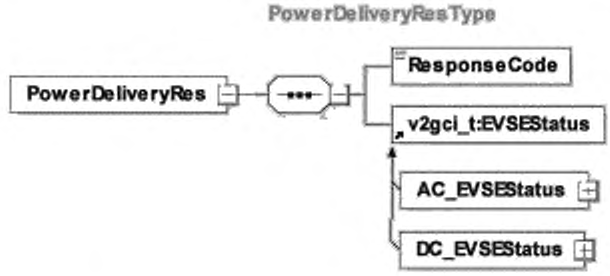


Рисунок 38 — Диаграмма — PowerDeliveryRes

[V2G2-224] SECC должен всегда согласовывать ChargingProfile EVCC (см. 8.5.2.10), если последний не превышает значений PMax элементов PMaxScheduleEntryType (см. 8.5.2.15) в соответствии с выбранным элементом SAScheduleTupleType (см. 8.5.2.13) в последнем сообщении ChargeParameterDiscoveryRes, отправленном SECC (см. 8.4.3.8.3).

[V2G2-777] Максимальное время задержки, которое дается EV для адаптации к новому номинальному уровню мощности (задний фронт) во время зарядки, составляет 5 с. Если время задержки превышено и мощность, отбираемая EV, превышает номинальное значение более чем на 10 %, EVSE может отключить EV.

[V2G2-225] SECC должен отправить отрицательный ResponseCode FAILED_ChargingProfileInvalid в сообщении-ответе PowerDelivery, если EVCC посылает ChargingProfile (см. 8.5.2.10), который не соответствует значениям PMax всех элементов PMaxScheduleEntryType (см. 8.5.2.15) согласно выбранному элементу SAScheduleTupleType (см. 8.5.2.13) в последнем сообщении ChargeParameterDiscoveryRes, отправленном SECC (см. 8.4.3.8.3).

[V2G2-226] Элементы данного сообщения должны использоваться, как определено в таблице 41.

Таблица 41 — Семантика и определение типа для PowerDeliveryRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
AC_EVSEStatus	complexType: AC_EVSEStatusType заменяет абстрактный тип EVSEStatusType см. 8.5.3.1	Этот элемент используется SECC для указания на статус EVSE и для сигнализации о событии, реакции на которое SECC ожидает от EVCC
DC_EVSEStatus	complexType: DC_EVSEStatusType заменяет абстрактный тип EVSEStatusType см. 8.5.4.1	Этот элемент используется SECC для указания на статус EVSE и для сигнализации о событии, реакции на которое SECC ожидает от EVCC

8.4.3.10 CertificateUpdateReq/Res

8.4.3.10.1 Обработка CertificateUpdateReq/Res

Обновление сертификата EVCC необходимо, когда срок его действия близится к завершению. В этом случае EVCC запрашивает новый сертификат от SECC для установки его в EVCC.

[V2G2-227] EVCC должен запросить необходимый сертификат у SECC при условии, что SECC предоставляет данную услугу, как ранее было указано в элементе ServiceList сообщения ServiceDiscoveryRes.

Примечание 1 — Если SECC не может предоставить запрошенный сертификат, то должна быть выполнена обработка ошибки, см. **[V2G2-471]**. Если EVSE не способно поддерживать данную функцию, оно должно быть идентифицировано, например, как «офлайновое EVSE».

Примечание 2 — Разрешается использовать обновление сертификата (как описано в данном пункте), только если цепь ContractSignatureCertChain не повреждена, т. е. она должна быть полностью действительна для выполнения данной процедуры. В противном случае должна быть использована такая же процедура, как для изначального сохранения сертификата контракта в EVCC. Один из вариантов выполнения процедуры — использование сообщений типа CertificateInstallationReq и CertificateInstallationRes.

8.4.3.10.2 CertificateUpdateReq

Отправляя сообщение CertificateUpdateReq, EV запрашивает SECC о представлении нового сертификата, который принадлежит действующему в текущее время контракту EV.

[V2G2-228] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 39.

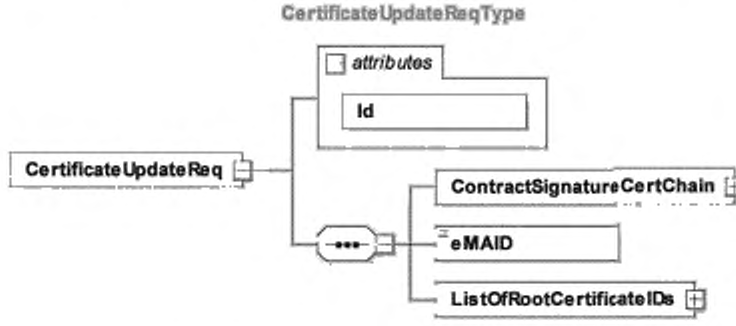


Рисунок 39 — Диаграмма — CertificateUpdateReq

[V2G2-229] Элементы данного сообщения должны использоваться, как определено в таблице 42.

[V2G2-888] EVCC должен подписать элемент тела CertificateUpdateReq, используя сертификат контракта, обновление которого запрашивается.

[V2G2-889] Служба выдачи сертификатов должна проверить подпись элемента CertificateUpdateReq, используя сертификат контракта, включенный в поле ContractSignatureCertChain. Служба выдачи сертификатов должна выполнить валидацию указанного сертификата, включая цепочку из того же поля, и убедиться в том, что он прослеживается до безопасного корневого удостоверяющего органа операторов услуг для EV. Если что-либо из вышеуказанного не выполняется, служба выдачи сертификатов должна рассматривать подпись как недействительную, прекратить процедуру и вызвать сообщение об ошибке от SECC.

Таблица 42 — Семантика и определение типа для CertificateUpdateReq

Имя элемента	Тип	Семантика
Id	simpleType: ID цепочка см. определение типа в F.6 (приложение F)	Этот элемент используется для ссылки на все тело сообщения в заголовке сообщения, когда необходимо применить подпись
ContractSignatureCertChain	complexType: CertificateChainType см. 8.5.2.5	Содержит имеющийся в настоящее время сертификат подписи, включая цепочку сертификата в EVCC. SECC использует этот сертификат (цепочку) для проверки подписи, включенной в заголовок сообщения. Передается полная цепочка, что обеспечивает возможность автономной проверки подписи
eMAID	simpleType: eMAIDType цепочка (мин. длина: 14, макс. длина: 15) см. определение типа в F.6 (приложение F)	Этот элемент идентифицирует контракт на зарядку. Формат определен в приложении G
ListOfRootCertificateIDs	complexType: ListOfRootCertificateIDsType см. 8.5.2.27	Этот список содержит идентификаторы сертификата всех корневых сертификатов, установленных в настоящее время в EVCC

8.4.3.10.3 CertificateUpdateRes

SECC после получения CertificateUpdateReq EVCC извлекает запрошенный сертификат у SA, поэтому необходимо установить онлайновое соединение. Затем SECC посылает EVCC CertificateUpdateRes, включая новый сертификат. В заключение EVCC устанавливает данный сертификат.

[V2G2-230] EVCC должен иметь сертификат и соответствующий закрытый ключ, пригодный для согласования подписи и ключа, с целью поддержки расшифровки закодированной информации.

[V2G2-231] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 40.

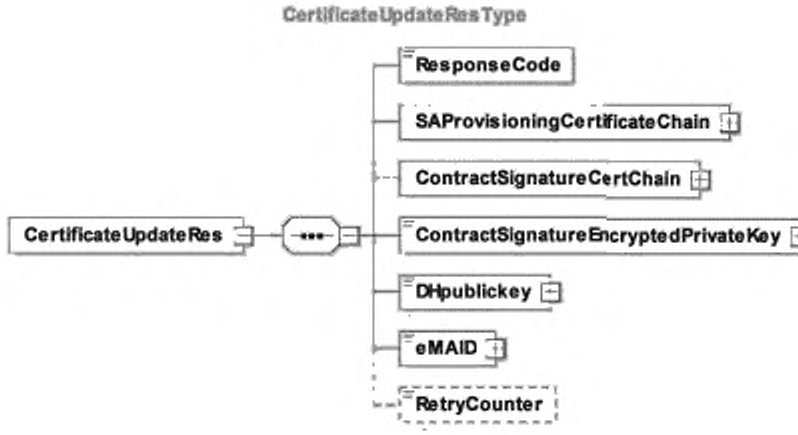


Рисунок 40 — Диаграмма — CertificateUpdateRes

[V2G2-232] Элементы данного сообщения должны использоваться, как определено в таблице 43.
[V2G2-928] Если ResponseCode в этом сообщении указывает на неудачу (т. е. начинается с FAILED). EVCC должен игнорировать все другие элементы, за исключением RetryCounter.

Таблица 43 — Семантика и определение типа для CertificateUpdateRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
SAProvisioningCertificate Chain	complexType: CertificateChainType см. 8.5.2.5	Переданная цепочка сертификата используется EVCC для проверки подписи в заголовке сообщения
ContractSignatureCert Chain	complexType: CertificateChainType см. 8.5.2.5	Новая цепочка сертификатов для подписей, которые должны быть установлены в EVCC
ContractSignature EncryptedPriva-teKey	complexType: ContractSignatureEncryptedPrivate KeyType см. 8.5.2.28	Закрытый ключ, который принадлежит новому сертификату для подписи. Он должен быть также установлен на EVCC. Эти данные должны быть зашифрованы с помощью старого сертификата контракта EVCC (на базе ContractSignatureCertChain, содержащегося в сообщении CertificateUpdateReq) и рассчитанного секретного ключа Диффи — Хеллмана для шифрования, как описано в 7.9.2.4.3
DHpublickey	complexType: DiffieHellmanPublickeyType см. определение типа в 8.5.2.29	Параметр открытого ключа Диффи — Хеллмана от SA для генерирования ключа сеанса EVCC с целью шифрования закрытого ключа подписи контракта на стороне SA и его дешифрования на стороне EVCC
eMAID	complexType: EMAIDType см. определение типа в 8.5.2.30	Этот элемент идентифицирует контракт на зарядку. Формат определен в приложении G

Окончание таблицы 43

Имя элемента	Тип	Семантика
RetryCounter	simpleType short см. определение типа в F.6 (приложение F)	Факультативно: если ResponseCode был «FAILED_NoCertificate Available» или «FAILED_ContractCanceled», то это поле содержит информацию о том, когда EVCC должен попытаться получить новый сертификат еще раз. Возможны следующие записи: x > 0: после «x» дней; 0: незамедлительно (при следующей зарядке); -1: никогда

[V2G2-233] SECC должен запросить необходимый сертификат у вторичного субъекта.

[V2G2-696] В случае ответа с информацией об ошибке сообщение должно также включать элемент RetryCounter.

Примечание 1 — SECC предлагает сервис по установке или обновлению сертификата, если он имеет возможность соединения с оператором услуг для EV, выпускающим сертификаты.

Примечание 2 — Если SECC не может предоставить запрошенный сертификат, то должна быть выполнена надлежащая обработка ошибки, см. **[V2G2-469]**. Если EVSE не способно поддерживать данную функцию, оно должно быть идентифицировано, например, как «офлайновое EVSE».

Примечание 3 — Разрешается использовать обновление сертификата, только если ContractSignatureCert не был испорчен, т. е. он должен быть полностью действителен для выполнения процедуры. В противном случае должна быть использована такая же процедура, как для изначального сохранения сертификата контракта в EVCC. Один из вариантов выполнения процедуры — использование сообщений типа CertificateInstallationReq и CertificateInstallationRes.

[V2G2-827] Сертификат контракта, предназначенный для замены установленного сертификата, должен быть пригоден для использования (в отношении срока действия) на EV сразу же после его передачи сообщением CertificateUpateRes.

Примечание 4 — Это необходимо для исключения необходимости для EVCC хранения и обращения с двумя сертификатами контракта.

[V2G2-890] Служба выдачи сертификатов должна подписать следующие элементы сообщения CertificateUpdateRes: ContractSignatureCertChain, ContractSignatureEncryptedPrivateKey, DHpublickey, eMAID. Сообщение должно включать цепочку сертификата для проверки этой подписи в поле SAProvisioningCertChain. Служба выдачи сертификатов должна убедиться в том, что данные, на которые распространяется эта подпись, получены от предшествующего SA подтвержденным способом.

[V2G2-891] EVCC должен проверить подпись сообщения CertificateUpdateRes, используя цепочку сертификатов подписчика SAProvisioningCertChain, проверить указанную цепочку и убедиться, что она вернулась к действительному корневому сертификату V2G. Он должен гарантировать, что сертификат подписчика имеет поле DC (DomainComponent) с содержимым «CPS». Он должен обеспечить включение в подпись следующих элементов: ContractSignatureCertChain, ContractSignatureEncryptedPrivateKey, DHpublickey, ContractID. EVCC должен подтвердить сертификат подписчика и убедиться, что он вернется к доверенному корневому сертификату V2G. Если что-либо из вышеперечисленного не выполняется, EVCC считает сообщение недействительным и отбрасывает его. Кроме того, EVCC должен хранить части цепочки сертификатов, которых она еще не имеет.

[V2G2-892] ContractSignatureCertChain, полученная в сообщении CertificateUpdateRes, должна быть целенаправленно сохранена таким образом, чтобы она могла быть применена позднее для проверки таблиц тарифных ставок. Сертификат Sub-CA2 оператора услуг для EV необходим для проверки таблиц тарифных ставок.

Примечание 5 — Не требуется, чтобы EVCC был способен проверить полученный им сертификат. Задача службы выдачи сертификатов — выдавать только действительные сертификаты.

8.4.3.11 CertificateInstallationReq/Res

8.4.3.11.1 Обработка CertificateInstallationReq/Res

Установка сертификата контракта в EVCC необходима, если EVCC в настоящее время не имеет действительного сертификата контракта, например не хранится ни один из сертификатов контракта, или у имеющихся сертификатов истек срок действия, или они отозваны. В этом случае EVCC запрашивает сертификат у SECC для установки в EVCC. Эта процедура, как правило, имеет место до начала процесса зарядки, потому что зарядка с авторизацией может начаться, только если у EVCC имеется действительный сертификат. SECC может запросить сертификат у SA. После установки данного сертификата может начинаться процесс зарядки на EVSE, к которому подключено EV.

Примечание 1 — Решение об использовании OEMProvisioningCert принимает изготовитель. Если OEMProvisioningCert в EVCC отсутствует, то эта процедура не может быть использована, а исходный сертификат контракта должен быть установлен в EVCC другим способом. Другие механизмы для установки сертификата контракта, помимо использования OEMProvisioningCert, не рассматриваются в настоящем стандарте.

[V2G2-234] SECC должен запросить необходимый сертификат у SA, если он имеет возможность подключения онлайн.

Примечание 2 — Если SECC не может предоставить запрошенный сертификат, то необходимо выполнить обработку ошибки. Если EVSE не поддерживает данную функцию, то оно должно быть идентифицировано, например, как «офлайновое EVSE». Если EVCC не имеет действующего сертификата, присвоенного контракту, режим «подключи и заряжай» не может использоваться до успешного выполнения процесса установки сертификата.

8.4.3.11.2 CertificateInstallationReq

С помощью сообщения CertificateInstallationReq EV запрашивает у SECC передачу сертификата, принадлежащего действующему контракту EV.

[V2G2-235] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 41.

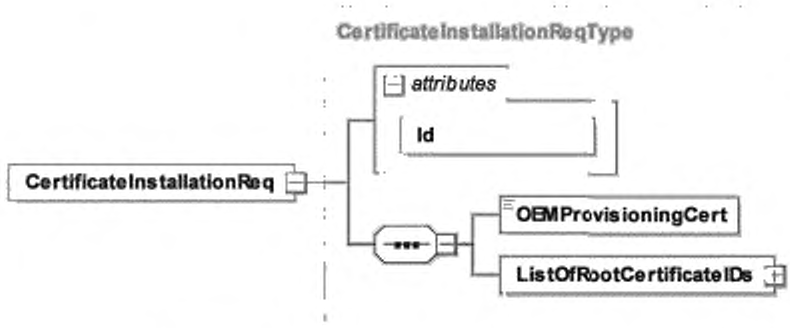


Рисунок 41 — Диаграмма — CertificateInstallationReq

[V2G2-236] Элементы данного сообщения должны использоваться, как определено в таблице 44.

Таблица 44 — Семантика и определение типа для CertificateInstallationReq

Имя элемента	Тип	Семантика
Id	simpleType: ID строка см. определение типа в F.6 (приложение F)	Этот элемент используется для ссылки на все тело сообщения в заголовке сообщения, когда необходимо применить подпись

Окончание таблицы 44

Имя элемента	Тип	Семантика
OEMProvisioningCert	simpleType: certificateType base64Binary (макс. длина 800) см. определение типа в F.6 (приложение F)	Конкретный сертификат EV, который был ранее установлен в EVCC, как правило, изготовителем. Идентификатор данного сертификата вместе с информацией, хранящейся у SA (контрактный партнер), используется для идентификации действующего контракта EV. Идентификатор сертификата выдается SA с использованием различных коммуникационных каналов. Сам сертификат (т. е. его открытый ключ) позднее используется для шифрования в CertificateInstallationRes. Сертификат закодирован в соответствии с особыми правилами кодирования DER (Distinguished Encoding Rules). См. также C.1 (приложение C)
ListOfRootCertificateIDs	complexType: ListOfRootCertificateIDsType см. 8.5.2.27	Этот список содержит идентификаторы сертификата всех корневых сертификатов, установленных в EVCC

[V2G2-893] EVCC должен подписать элемент тела CertificateInstallationReq, используя свой сертификат изготовителя.

[V2G2-894] Служба выдачи сертификатов должна проверить подпись элемента тела сообщения CertificateInstallationReq с помощью включенного сертификата изготовителя из поля OEMProvisioningCert. Служба выдачи сертификатов должна выполнить валидацию указанного сертификата изготовителя, включая цепочку из того же поля, и убедиться в том, что он прослеживается до безопасного корневого сертификата изготовителя. Служба выдачи сертификатов должна убедиться в том, что сертификат изготовителя имеет установленный DC (доменный компонент) «OEM». Если что-либо из вышеуказанного не выполняется, служба выдачи сертификатов должна рассматривать подпись как недействительную, прекратить процедуру и вызвать сообщение об ошибке от SECC.

8.4.3.11.3 CertificateInstallationRes

После получения от EVCC CertificateInstallationReq SECC направляет CertificateInstallationRes, включая запрошенный сертификат. Затем EVCC устанавливает данный сертификат.

В EVCC предоставляется только один сертификат: сертификат для подписания сообщений. Он принадлежит действующему в настоящее время контракту EV.

[V2G2-237] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 42.

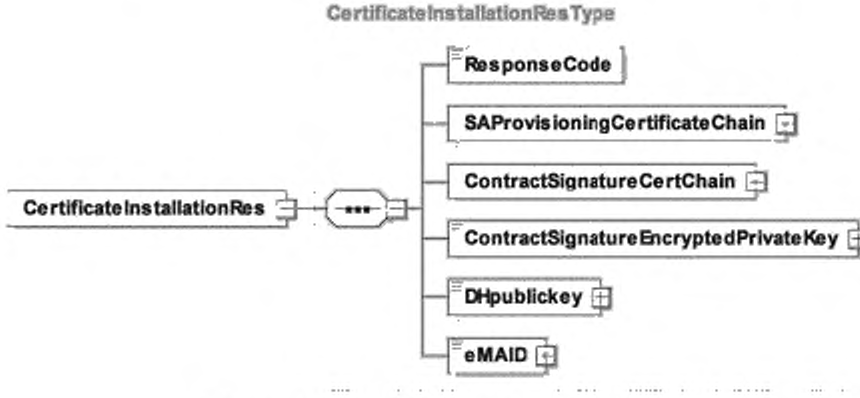


Рисунок 42 — Диаграмма — CertificateInstallationRes

[V2G2-238] Элементы данного сообщения должны использоваться, как определено в таблице 45.

Таблица 45 — Семантика и определение типа для CertificateInstallationRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F.6)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
SAProvisioningCertificate Chain	complexType: CertificateChainType см. 8.5.2.5	Переданная цепочка сертификата используется EVCC для проверки подписи в заголовке сообщения
ContractSignatureCert Chain	complexType: CertificateChainType см. 8.5.2.5	Новая цепочка сертификата для подписи, которая должна быть установлена в EVCC
ContractSignature EncryptedPrivateKey	complexType: ContractSignatureEncryptedPrivate KeyType см. 8.5.2.28	Закрытый ключ, который принадлежит новому сертификату для подписи. Он должен быть также установлен в EVCC. Эти данные должны быть зашифрованы с помощью предоставленного сертификата изготовителя EVCC (на базе OEMProvisioningCert, содержащегося в сообщении Certificate Installation Request) и с помощью секретного ключа Диффи — Хеллмана для шифрования, как описано в 7.9.2.4.3
DHpublickey	complexType: DiffieHellmanPublickeyType см. определение типа в 8.5.2.29	Открытый ключ Диффи — Хеллмана от SA для генерирования ключа сеанса на стороне EVCC с целью дальнейшего шифрования закрытого ключа подписи контракта на стороне SA и дешифровывания его на стороне EVCC
eMAID	complexType: EMAIDType см. определение типа в 8.5.2.30	Этот элемент идентифицирует контракт на зарядку. Формат определен в приложении G

[V2G2-895] Служба выдачи сертификатов должна подписать следующие элементы сообщения CertificateInstallationRes с использованием сертификата службы выдачи сертификатов: ContractSignatureCertChain, ContractSignatureEncryptedPrivateKey, DHpublickey, ContractID. CertificateInstallationRes должно включать цепочку сертификата для проверки этой подписи в поле SAProvisioningCertChain. Служба выдачи сертификатов также должна убедиться в том, что данные, которые удостоверяются этой подписью, получены от предыдущего SA надежным способом.

[V2G2-896] EVCC должен проверить подпись сообщения CertificateInstallationRes, используя цепочку сертификатов подписчика SAProvisioningCertificateChain, проверить указанную цепочку и убедиться, что она вернулась к действительному корневому сертификату V2G. Он должен гарантировать, что сертификат подписчика имеет поле DC (DomainComponent) с содержимым «CPS». Он должен обеспечить включение в подпись следующих элементов: ContractSignatureCertChain, ContractSignatureEncrypted PrivateKey, DHpublickey, ContractID. Если какое-либо из вышеперечисленных условий не выполняется, EVCC считает сообщение недействительным и отбрасывает его. Кроме того, EVCC должен хранить части цепочки сертификатов, которых она еще не имеет.

[V2G2-897] ContractSignatureCertChain, полученная в сообщении CertificateInstallationRes, должна быть сохранена таким образом, чтобы она могла быть применена позднее для проверки таблиц тарифных ставок. Сертификат Sub-CA 2 оператора услуг для EV необходим для проверки таблиц тарифных ставок.

Примечание — Не требуется, чтобы EVCC мог проверить полученный им сертификат. Задача службы выдачи сертификатов — выдавать только действительные сертификаты.

8.4.3.11.4 Офлайновая установка сертификата

В качестве альтернативы описанной ранее процедуре сертификат контракта может быть передан на транспортное средство без использования протокола зарядки. Это может быть необходимо, например, если инфраструктура, SA или транспортное средство не поддерживает CertificateInstallationReq/Response. В таких случаях сертификат контракта, который должен быть установлен, передается потребителю (например, почтой, электронной почтой) и затем EV (например, с использованием диагностического интерфейса транспортного средства или интернет-доступа к транспортному средству). Это означает, что все указанные передачи осуществляются в офлайновом режиме, а не через протокол зарядки. Файл (передаваемый потребителю) содержит цепочку сертификата и соответствующий закрытый ключ (аналогично CertificateInstallationRes). Чтобы избежать несовместимости форматов файлов и необходимости реализации алгоритмов трансформации, единообразные форматы файлов должны использоваться всеми SA при рассылке файлов сертификата контракта.

[V2G2-648] При передаче SA файла, который содержит сертификат контракта, цепочку сертификата и закрытый ключ подписи, файл должен быть предоставлен в формате PKCS#12.

Примечание — Решение о шифровании данных, содержащихся в данном файле, принимается SA. Если данные шифруются, потребителю должен быть дополнительно передан пароль. Для передачи пароля должен использоваться безопасный канал. Как альтернатива, данные, содержащиеся в файле PKCS#12, могут не шифроваться, а передаваться лично безопасным путем (например, вручаться лично).

8.4.3.12 SessionStopReq/Res

8.4.3.12.1 SessionStopReq/Res Handling

Эта пара сообщений V2G должна использоваться для прекращения сеанса связи V2G, инициированного предшествующим сообщением SessionSetupReq.

8.4.3.12.2 SessionStopReq

С помощью сообщения SessionStopReq EVCC запрашивает о разрешении прекращения процесса зарядки.

[V2G2-239] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 43.



Рисунок 43 — Диаграмма — SessionStopReq

[V2G2-738] Элементы данного сообщения должны использоваться, как определено в таблице 46.

Таблица 46 — Семантика и определение типа для SessionStopReq

Имя элемента	Тип	Семантика
ChargingSession	simpleType: ChargingSessionType enumeration см. определение типа в F.6 (приложение F)	ChargingSession указывает на намерение EVCC сделать паузу или прекратить сеанс связи V2G

8.4.3.12.3 SessionStopRes

SECC после получения SessionStopReq от EVCC посылает SessionStopRes, информируя EVCC об успешном прекращении процесса зарядки.

[V2G2-240] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 44.

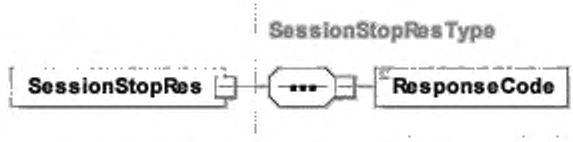


Рисунок 44 — Диаграмма — SessionStopRes

[V2G2-241] Элементы данного сообщения должны использоваться, как определено в таблице 47.

Таблица 47 — Семантика и определение типа для SessionStopRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC

8.4.3.13 MeteringReceiptReq/Res

8.4.3.13.1 MeteringReceiptReq/Res Handling

При отправлении сообщения MeteringReceiptReq EVCC подтверждает, что элементы данных MeterInfo, SessionID и SAScheduleTupleID, включенные в сообщение ChargingStatusRes до этого запроса, были получены от SECC. Это подтверждение реализуется подписью к телу сообщения MeteringReceiptReq. Подпись, применяемая EVCC, предназначена только для подтверждения, что EVCC вместе с действующим сертификатом контракта на зарядку, который был выбран для данного сеанса зарядки, получил элементы данных MeterInfo record, SessionID и SAScheduleTupleID в составе сообщения Charging Status Res до данного запроса. Она не предназначена для подтверждения, что количество энергии, указанное в предыдущем MeterInfo record, верно. Однако подписанная информация о показаниях счетчика может быть использована любым оператором услуг для EV для расчета. Определение процесса расчета, который подчиняется локальным правилам, не рассматривается в настоящем стандарте.

[V2G2-902] Элемент MeterInfo, отправленный SECC в сообщении ChargingStatusRes, должен содержать такие же данные, какие выдает прибор учета. Он должен содержать необработанные показания прибора учета в форме, которая в целом поддается машинному чтению.

Примечание — Это позволит EVCC в общем случае анализировать показания прибора учета.

8.4.3.13.2 MeteringReceiptReq

При отправлении MeteringReceiptReq EVCC подтверждает, что элементы данных MeterInfo record, SessionID и SAScheduleTupleID были получены от SECC. Это подтверждение реализуется путем применения подписи к телу сообщения MeteringReceiptReq.

[V2G2-245] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 45.

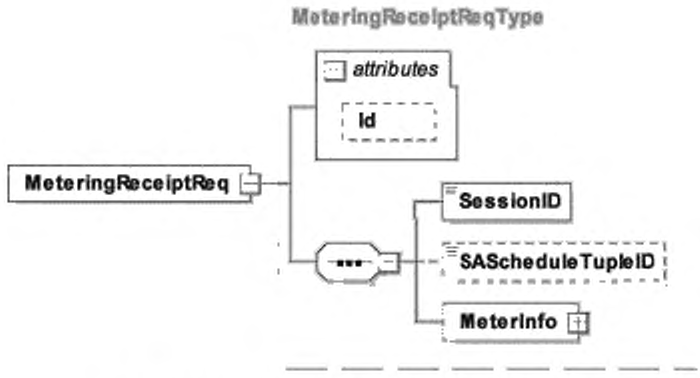


Рисунок 45 — Диаграмма — MeteringReceiptReq

[V2G2-246] Элементы данного сообщения должны использоваться, как определено в таблице 48.

Таблица 48 — Семантика и определение типа для MeteringReceiptReq

Имя элемента	Тип	Семантика
Id	simpleType: ID цепочка см. определение типа в F.6 (приложение F)	Факультативно: этот атрибут используется для ссылки на тело сообщения в заголовке подписи в случае применения подписи
SessionID	simpleType: SessionIDType SessionIDType: hexBinary (макс. длина: 8) см. определение типа в F.6 (приложение F)	Этот элемент сообщения используется EVCC и SECC для уникальной идентификации сеанса связи V2G. Этот элемент идентичен элементу, включенному в заголовок сообщения. Он помещается в тело дополнительно, чтобы была возможность применить к нему подпись (подписывается все тело сообщения)
SAScheduleTupleID	simpleType: SAIDType short см. определение типа в F.6 (приложение F)	Факультативно: уникальный идентификатор элемента SAScheduleTuple во время сеанса зарядки. Этот элемент является лишь повторением значения, полученного от SECC либо в ChargingStatusRes (во время зарядки переменным током), либо в сообщении CurrentDemandRes (во время зарядки постоянным током). Идентификатор SA (SAID) остается уникальным идентификатором для одного графика в течение сеанса зарядки
MeterInfo	complexType: MeterInfoType см. 8.5.2.6	Если SECC указал в ChargingStatusRes (зарядка переменным током)/CurrentDemandRes (зарядка постоянным током), что MeteringReceiptReq необходим, то этот элемент сообщения является повторением элемента MeterInfo, полученного от SECC в ChargingStatusRes (зарядка переменным током)/CurrentDemandRes (зарядка постоянным током)

[V2G2-776] Элемент сообщения SAScheduleTupleID, если он включен, должен быть всегда равен значению SAScheduleTupleID, полученному в предыдущем сообщении ChargingStatusRes (зарядка переменным током)/CurrentDemandRes (зарядка постоянным током).

[V2G2-903] EVCC должен подписать тело сообщения MeteringReceiptReq с помощью закрытого ключа, принадлежащего сертификату контракта, который он отправил в данном сеансе в сообщении PaymentDetailsReq.

[V2G2-904] SECC/SA может проверить подпись элемента тела сообщения MeteringReceiptReq, используя сертификат контракта, который он получил в данном сеансе в составе сообщения PaymentDetailsReq. Если результат проверки отрицателен, квитанция прибора учета должна рассматриваться как недействительная.

Примечание 1 — SECC может представить сертификат контракта для SA вместе с подписанной MeteringReceipt, с тем чтобы SA мог позднее перепроверить квитанцию. В случае, если SA уже сохранил сертификат контракта, то идентификатора и SA, выдавшего сертификат контракта, может быть достаточно.

Примечание 2 — EVCC может быть не способен проверить подпись SigMeterReading, которая была сгенерирована SECC и которая включена в элемент MeterInfo, например, потому что некоторые данные, необходимые для подписи, не переданы. Поэтому применяемая EVCC подпись, которая описывается в настоящем пункте, служит только в качестве подтверждения, что EVCC получил эти данные, и не составляет какого-либо подтверждения корректности.

8.4.3.13.3 MeteringReceiptRes

SECC после получения MeteringReceiptReq от EVCC посылает MeteringReceiptRes, информируя EVCC о том, была ли квитанция успешно получена и принята SECC.

[V2G2-247] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 46.

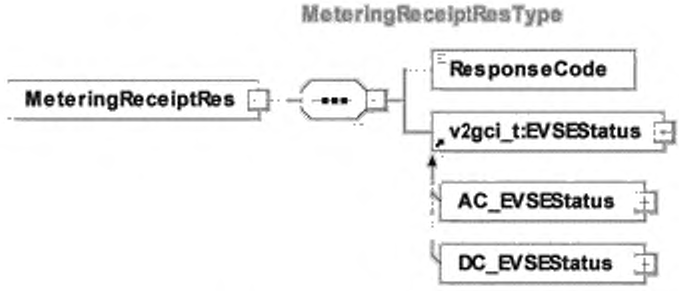


Рисунок 46 — Диаграмма — MeteringReceiptRes

[V2G2-248] Элементы данного сообщения должны использоваться, как определено в таблице 49.

Таблица 49 — Семантика и определение типа для MeteringReceiptRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
AC_EVSEStatus	complexType: AC_EVSEStatusType заменяет абстрактный тип EVSEStatusType см. 8.5.3.1	Этот элемент используется SECC для указания на статус EVSE и для сигнализации о событии, реакции на которое SECC ожидает от EVCC
DC_EVSEStatus	complexType: DC_EVSEStatusType заменяет абстрактный тип EVSEStatusType см. 8.5.4.1	Этот элемент используется SECC для указания на статус EVSE и для сигнализации о событии, реакции на которое SECC ожидает от EVCC

8.4.4 Сообщения для зарядки переменным током

8.4.4.1 Обзор

Сообщения, описываемые как сообщения для зарядки переменным током, принадлежат к набору(ам) сообщений для зарядки переменным током.

8.4.4.2 ChargingStatusReq/Res

8.4.4.2.1 Обработка ChargingStatusReq/Res

Пара сообщений Charging Status обеспечивает проверку работоспособности по показаниям прибора учета, предоставляемых SECC. На основе повторяемого обмена сообщениями Charging Status EV имеет средство проверки и валидации мощности, отобранной у EVSE. Это также позволяет SECC запросить EVCC о подписании информации прибора учета, включенной в сообщение ChargingStatusRes с использованием пары сообщений-квитанций прибора учета. Использование данной подписанной информации прибора учета для расчета может быть предметом регулирования в определенных странах. Дополнительно сообщение-запрос используется для продолжения сеанса связи (сведения об обработке сеанса см. в 8.7.2).

8.4.4.2.2 ChargingStatusReq

[V2G2-242] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 47.

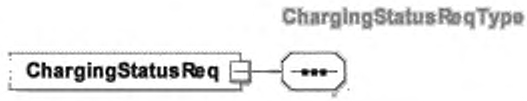


Рисунок 47 — Диаграмма — ChargingStatusReq

8.4.4.2.3 ChargingStatusRes

SECC после получения ChargingStatusReq от EVCC посылает ChargingStatusRes. В случае успешного запроса статуса прибора учета (Metering Status Request) ответ предоставляет текущие показания прибора учета, установленного на EVSE. В случае неудачного считывания показаний прибора учета показание прибора не выдается. На неудачу указывает ResponseCode.

[V2G2-243] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 48.

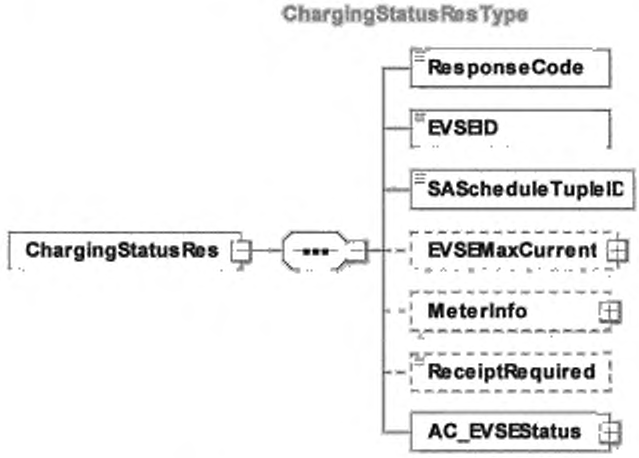


Рисунок 48 — Диаграмма — ChargingStatusRes

[V2G2-244] Элементы данного сообщения должны использоваться, как определено в таблице 50.

Таблица 50 — Семантика и определение типа для ChargingStatusRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
EVSEID	simpleType: evseIDType строка (мин. длина: 7, макс. длина: 37) см. определение типа в F.6 (приложение F)	Любой идентификатор, который уникально идентифицирует EVSE и точку заряда, к которой подключено транспортное средство. Формат этого элемента сообщения определен в приложении G. Если SECC не может предоставить такие идентификационные данные, то значение EVSEID устанавливается на ноль («ZZ00000»)
SAScheduleTupleID	simpleType: SAIDType short см. определение типа в F.6 (приложение F)	Уникальный идентификатор в течение сеанса зарядки для элемента SAScheduleTuple. Используется SECC для указания, какой тариф применяется в настоящее время. Идентификатор SA (SAID) остается уникальным идентификатором для одного графика в течение сеанса зарядки
EVSEMaxCurrent	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: этот элемент используется SECC для индикации максимального фазного тока по фазам, который EV может отбирать. Этот элемент не включается в сообщение, если был выбран какой-либо набор сообщений для зарядки переменным током в режиме PnC
MeterInfo	complexType: MeterInfoType см. 8.5.2.6	Факультативно: включает запись MeterInfo, содержащую последнее показание прибора учета и другие значимые для учета данные
ReceiptRequired	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Факультативно: данный элемент используется SECC для указания на то, что SECC запрашивает отправку сообщения MeteringReceiptReq с целью подписания информации прибора учета. Если ReceiptRequired равно True, от SECC запрашивается отправка сообщения MeteringReceiptReq, включая подпись. Если ReceiptRequired равно False, от EVCC не требуется отправлять сообщение MeteringReceiptReq
AC_EVSEStatus	complexType: AC_EVSEStatusType заменяет абстрактный тип EVSEStatusType см. 8.5.3.1	Этот элемент используется SECC для указания своего статуса

8.4.5 XML-сообщения

8.4.5.1 Обзор

Сообщения, определяемые как сообщения для зарядки постоянным током (DC-Messages), принадлежат к набору(ам) сообщений для зарядки постоянным током (см. 8.6.2.4).

8.4.5.2 CableCheckReq/Res

8.4.5.2.1 Обработка CableCheckReq/Res

Для безопасной зарядки постоянным током должна быть выполнена проверка кабеля.

8.4.5.2.2 CableCheckReq

CableCheckReq запрашивает статус проверки кабеля EVSE и, например, сообщает EVSE, зафиксирован ли кабель на стороне EV и о готовности EV к зарядке.

[V2G2-249] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 49.

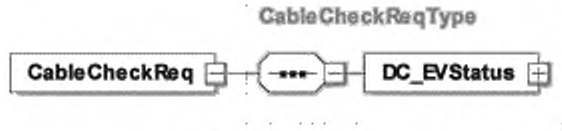


Рисунок 49 — Диаграмма — CableCheckReq

[V2G2-250] Элементы данного сообщения должны использоваться, как определено в таблице 51.

Таблица 51 — Семантика и определение типа для CableCheckReq

Имя элемента	Тип	Семантика
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока зарядки EV

8.4.5.2.3 CableCheckRes

SECC после получения CableCheckReq от EVCC посылает CableCheckRes, информируя EVCC о результатах проверки кабеля и статусе EVSE.

[V2G2-251] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 50.

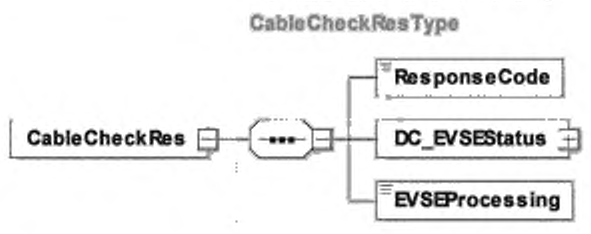


Рисунок 50 — Диаграмма — CableCheckRes

[V2G2-252] Элементы данного сообщения должны использоваться, как определено в таблице 52.

Таблица 52 — Семантика и определение типа для CableCheckRes

Имя элемента	Тип	Семантика
EVSEProcessing	simpleType: EVSEProcessingType enumeration см. определение типа в F.6 (приложение F)	Параметр, показывающий, что EVSE завершило обработку, которая была инициирована CableCheckReq, или если EVSE еще продолжает обработку в это время, что сообщение-ответ было отправлено
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC

Окончание таблицы 52

Имя элемента	Тип	Семантика
DC_EVSEStatus	complexType: DC_EVSEStatusType см. 8.5.4.1	Актуальное значение тока зарядки EVSE

Примечание — С помощью параметра EVSEProcessing EVSE может показать EVCC, что обработка не завершилась, но сообщение-ответ должно быть отправлено для выполнения требований по тайм-ауту и производительности, изложенных в 8.7.2. Это позволяет продолжить сеанс связи с выполнением требований по производительности и тайм-ауту.

8.4.5.3 PreChargeReq/Res

8.4.5.3.1 Обработка PreChargeReq/Res

Предзарядка используется для адаптации выходного напряжения EVSE к напряжению аккумуляторной батареи EV.

8.4.5.3.2 PreChargeReq

PreChargeReq используется для запуска процесса предзарядки со стороны EV.

[V2G2-253] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 51.

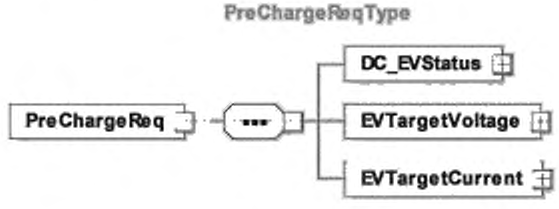


Рисунок 51 — Диаграмма — PreChargeReq

[V2G2-254] Элементы данного сообщения должны использоваться, как определено в таблице 53.

Таблица 53 — Семантика и определение типа для PreChargeReq

Имя элемента	Тип	Семантика
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока зарядки EV
EVTargetVoltage	complexType: PhysicalValueType см. 8.5.2.7	Целевое напряжение, запрошенное EV
EVTargetCurrent	complexType: PhysicalValueType см. 8.5.2.7	Ток, запрошенный EV

8.4.5.3.3 PreChargeRes

SECC после получения PreChargeReq от EVCC посылает PreChargeRes, информируя EV о статусе EVSE и текущем выходном напряжении EVSE.

[V2G2-255] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 52.

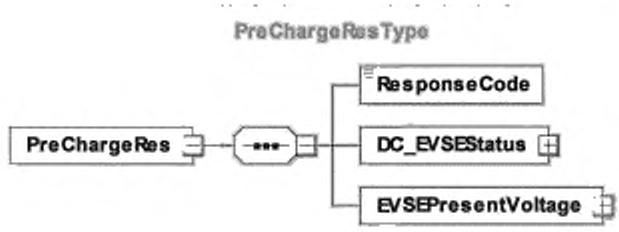


Рисунок 52 — Диаграмма — PreChargeRes

[V2G2-256] Элементы данного сообщения должны использоваться, как определено в таблице 54.

Таблица 54 — Семантика и определение типа для PreChargeRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
DC_EVSEStatus	complexType: DC_EVSEStatusType см. 8.5.4.1	Актуальное значение тока зарядки EVSE
EVSEPresentVoltage	complexType: PhysicalValueType см. 8.5.2.7	Выходное напряжение EVSE по *[9]*

8.4.5.4 CurrentDemandReq/Res

8.4.5.4.1 Обработка CurrentDemandReq/Res

Для управления зарядкой постоянным током EV требуется периодически передавать запрашиваемый ток. Также передаются целевое напряжение и разница (между целевыми и фактическими значениями) тока и напряжения.

8.4.5.4.2 CurrentDemandReq

С помощью сообщения CurrentDemandReq EV запрашивает определенный ток у EVSE. Также передаются целевое напряжение и разница (между целевыми и фактическими значениями) тока и напряжения.

[V2G2-257] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 53.

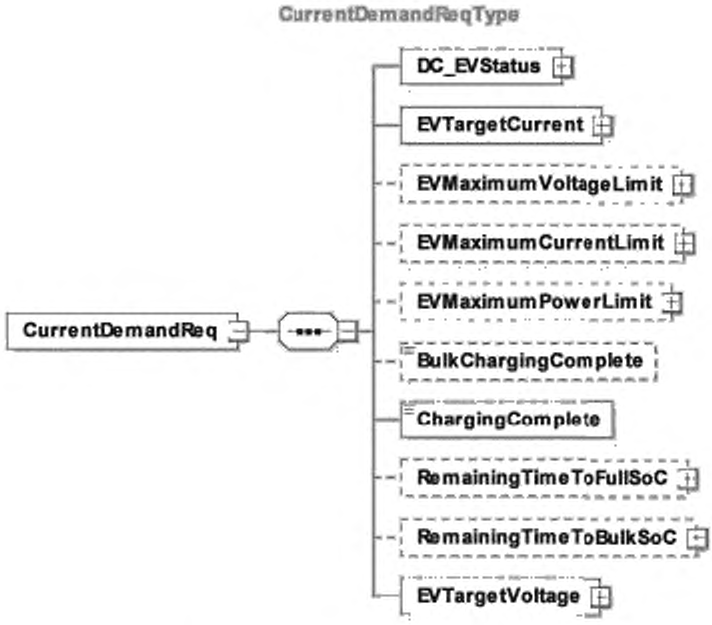


Рисунок 53 — Диаграмма — CurrentDemandReq

[V2G2-258] Элементы данного сообщения должны использоваться, как определено в таблице 55.

Таблица 55 — Семантика и определение типа для CurrentDemandReq

Имя элемента	Тип	Семантика
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока зарядки EV
EVTargetCurrent	complexType: PhysicalValueType см. 8.5.2.7	Ток, запрашиваемый EV в данный момент времени
EVMaximumVoltageLimit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальное напряжение, допускаемое EV
EVMaximumCurrentLimit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальный ток, допускаемый EV
EVMaximumPowerLimit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальная мощность, допускаемая EV
BulkChargingComplete	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Факультативно: если установлено значение True, то EV сообщает, что основная зарядка (степень заряда около 80 %) завершена
ChargingComplete	simpleType: логическое выражение см. определение типа в F.6 (приложение F)	Если установлено значение True, EV сообщает, что полная зарядка (степень заряда 100 %) завершена

Окончание таблицы 55

Имя элемента	Тип	Семантика
RemainingTimeToFullSoC	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: оценочное или рассчитанное время до выполнения полной зарядки (степень заряда 100 %)
RemainingTimeToBulkSoC	complexType PhysicalValueType см. 8.5.2.7	Факультативно: оценочное или рассчитанное время до выполнения основной зарядки (степень заряда 80 %)
EVTargetVoltage	complexType PhysicalValueType см. 8.5.2.7	Целевое напряжение, запрашиваемое EV

8.4.5.4.3 CurrentDemandRes

SECC после получения CurrentDemandReq от EVCC посылает CurrentDemandRes, информируя EV о статусе EVSE и текущих выходных значениях напряжения и тока.

[V2G2-259] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 54.

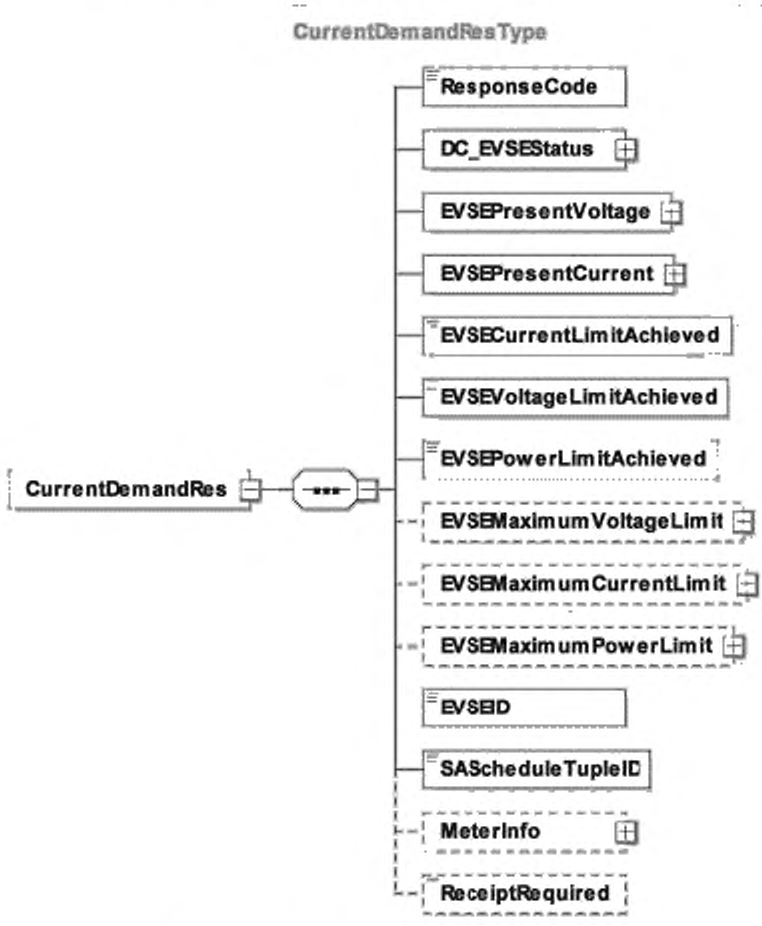


Рисунок 54 — Диаграмма — CurrentDemandRes

[V2G2-260] Элементы данного сообщения должны использоваться, как определено в таблице 56.

Таблица 56 — Семантика и определение типа для CurrentDemandRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
DC_EVSEStatus	complexType: DC_EVSEStatusType см. 8.5.4.1	Актуальное значение тока зарядки EVSE
EVSEPresentVoltage	complexType: PhysicalValueType см. 8.5.2.7	Текущее выходное напряжение EVSE. См. таблицу сопоставления SAE — ИСО*
EVSEPresentCurrent	complexType: PhysicalValueType см. 8.5.2.7	Текущий выходной ток EVSE. См. таблицу сопоставления SAE — ИСО*
EVSECurrentLimit Achieved	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Если установлено значение True, EVSE достигло своего предела по току
EVSEVoltageLimit Achieved	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Если установлено значение True, EVSE достигло своего предела по напряжению
EVSEPowerLimit Achieved	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Если установлено значение True, EVSE достигло своего предела по мощности
EVSEMaximumVoltage Limit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальное напряжение EVSE
EVSEMaximumCurrent Limit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальный ток EVSE
EVSEMaximumPower Limit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальная мощность EVSE
EVSEID	simpleType: evseIDType SessionIDType: hexBinary (макс. длина: 32) см. определение типа в F.6 (приложение F)	Любой идентификатор, уникальным образом идентифицирующий EVSE и точку заряда, к которым подключено транспортное средство. Формат этого элемента сообщений определен в G.2 (приложение G). Если SECC не может предоставить такие идентификационные данные, значение EVSEID устанавливается на ноль (00hex)
SAScheduleTupleID	simpleType: SAIDType short см. определение типа в F.6 (приложение F)	Уникальный идентификатор элемента SAScheduleTuple на протяжении сеанса зарядки. Используется SECC для указания тарифа который применяется в настоящее время. Идентификатор SA (SAID) остается уникальным идентификатором для одного графика в течение сеанса зарядки

Окончание таблицы 56

Имя элемента	Тип	Семантика
MeterInfo	complexType: MeterInfoType см. 8.5.2.6	Факультативно: включает запись MeterInfo, содержащую последнее показание прибора учета и другие значимые для учета данные
ReceiptRequired	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Факультативно: этот элемент используется SECC для отправки сообщения MeteringReceiptReq с целью подписания записи прибора учета. Если ReceiptRequired равно True, от EVCC запрашивается отправка сообщения MeteringReceiptReq, включая подпись. Если ReceiptRequired равно False, от EVCC не требуется отправлять сообщение MeteringReceiptReq
* *Таблица сопоставления SAE — ИСО приведена в приложении I.*		

8.4.5.5 WeldingDetectionReq/Res

8.4.5.5.1 Обработка WeldingDetectionReq/Res

Сообщения об определении сваривания контактов, описанные в настоящем пункте, позволяют реализовать механизм определения сваривания контактов, как описано в *[9]*.

8.4.5.5.2 WeldingDetectionReq

С помощью сообщения WeldingDetectionReq EV запрашивает определение сваривания контактов на EVSE.

[V2G2-261] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 55.

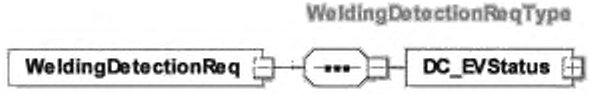


Рисунок 55 — Диаграмма — WeldingDetectionReq

[V2G2-262] Элементы данного сообщения должны использоваться, как определено в таблице 57.

Таблица 57 — Семантика и определение типа для WeldingDetectionReq

Имя элемента	Тип	Семантика
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока зарядки EV

8.4.5.5.3 WeldingDetectionResponse

SECC после получения WeldingDetectionReq от EVCC посылает Welding DetectionResponse, информируя EV о статусе EVSE и текущем выходном напряжении EVSE.

[V2G2-263] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать обязательные сообщения и элементы сообщений, как определено в таблице 104 и в соответствии с рисунком 56.

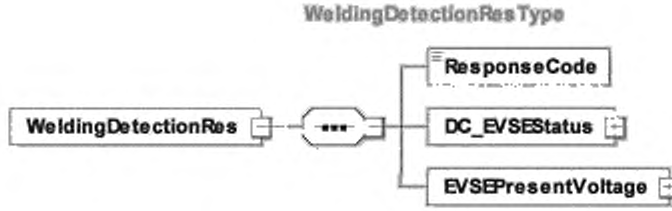


Рисунок 56 — Диаграмма — WeldingDetectionRes

[V2G2-264] Элементы данного сообщения должны использоваться, как определено в таблице 58.

Таблица 58 — Семантика и определение типа для WeldingDetectionRes

Имя элемента	Тип	Семантика
ResponseCode	simpleType: responseCodeType enumeration см. определение типа в F.6 (приложение F)	ResponseCode, показывающий статус подтверждения любого из сообщений V2G, полученных SECC
DC_EVSEStatus	complexType: DC_EVSEStatusType см. 8.5.4.1	Актуальное значение тока зарядки EVSE
EVSEPresentVoltage	complexType: PhysicalValueType см. 8.5.2.7	Текущее напряжение EVSE, см. выходное напряжение SAE

8.5 Комплексные типы данных

8.5.1 Обзор

В данном подразделе описываются комплексные типы данных, которые используются в сообщениях. Комплексные типы данных состоят из нескольких элементов, которые сами основаны на простых типах данных.

8.5.2 Общие сведения

8.5.2.1 ServiceType

Данный тип представляет собой признак конкретного сервиса. Он дает короткое определение и идентификацию конкретному сервису.

[V2G2-265] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 57.

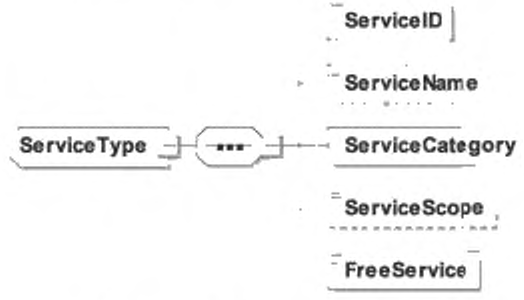


Рисунок 57 — Диаграмма — ServiceType

[V2G2-266] Элемент сообщения должен использоваться, как определено в таблице 59.

Таблица 59 — Семантика и определение типа для ServiceType

Имя элемента	Тип	Семантика
ServiceID	simpleType: unsignedShort см. определение типа в F.6 (приложение F)	Уникальный идентификатор сервиса
ServiceName	simpleType: serviceNameType цепочка (макс. длина: 32) см. определение типа в F.6 (приложение F)	Факультативно: удобочитаемое имя сервиса
ServiceCategory	simpleType: serviceCategoryType enumeration см. определение типа в F.6 (приложение F)	Категория сервиса соответствует определенным сервисам, полученным из базовых сервисов
ServiceScope	simpleType: serviceScopeType цепочка (макс. длина: 64) см. определение типа в F.6 (приложение F)	Факультативно: дополнительная информация об использовании данного сервиса
FreeService	simpleType логическое выражение см. определение типа в F.6 (приложение F)	Элемент используется SECC для указания, может или нет EVCC использовать данный сервис бесплатно. Если FreeService равно True, EV может использовать предлагаемый сервис бесплатно. Если FreeService равно False, сервис, если он используется EV, будет оплачиваться с помощью метода оплаты, согласованного с помощью элемента сообщения PaymentOptionListType

8.5.2.2 ServiceListType

[V2G2-267] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 58.



Рисунок 58 — Диаграмма — ServiceListType

[V2G2-268] Элемент сообщения должен использоваться, как определено в таблице 60.

Таблица 60 — Семантика и определение типа для ServiceListType

Имя элемента	Тип	Семантика
Service	complexType: serviceType	Содержит всю информацию для идентификации сервиса. ServiceList включает не менее одного сервиса и может включать до восьми сервисов

8.5.2.3 ChargeServiceType

Данный тип — это конкретный сервис по зарядке EV, производный от ServiceType (см. 8.5.2.1), содержащий дополнительную информацию о SupportedEnergyTransferMode(s), предлагаемых EVSE.

[V2G2-271] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 59.

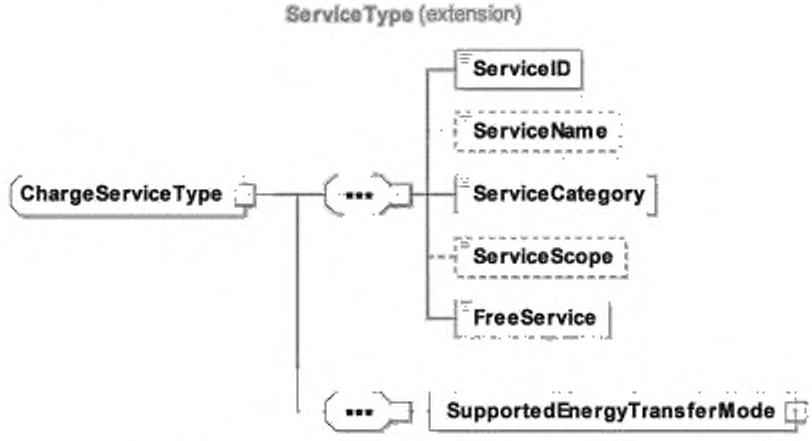


Рисунок 59 — Диаграмма — ChargeServiceType

[V2G2-272] Элемент сообщения должен использоваться, как определено в таблице 61.

Таблица 61 — Семантика и определение типа для ChargeServiceType

Имя элемента	Тип	Семантика
ServiceType (extension)	complexType: ServiceType см. определение типа в 8.5.2.1	Этот набор элементов сообщений перечислен в ServiceType. См. описание семантики данных элементов сообщений в таблице 59
SupportedEnergyTransferMode	complexType см. определение типа в 8.5.2.4	Имеющиеся режимы передачи энергии, которые поддерживает EVSE

Определение и использование EnergyTransferModeType в SupportedEnergyTransferModeType поддерживают разъемы типа 1, 2 и 3 в соответствии с *ГОСТ Р МЭК 62196-2* и разъемы в соответствии с *[49]*.

8.5.2.4 SupportedEnergyTransferModeType

[V2G2-757] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 62 и в соответствии с рисунком 60.



Рисунок 60 — Диаграмма — SupportedEnergyTransferModeType

[V2G2-758] Элемент сообщения должен использоваться, как определено в таблице 62.

Таблица 62 — Семантика и определение типа для SupportedEnergyTransferModeType

Имя элемента	Тип	Семантика
EnergyTransferMode	simpleType: EnergyTransferModeType enumeration см. определение типа в F.6 (приложение F) и таблице 63	Режим передачи энергии для зарядки, который поддерживается SECC. См. подробности в таблице 63. Максимальное вхождение 6

[V2G2-759] EVCC и SECC должны использовать EnergyTransferModeType, как описано в таблице 63.

Таблица 63 — Семантика для EnergyTransferModeType

EnergyTransferMode	Предлагаемый зарядный сервис
AC_single_phase_core	Зарядка однофазным переменным током в соответствии с *ГОСТ Р МЭК 62196-1*, *ГОСТ Р МЭК 62196-2* и *[49]*
AC_three_phase_core	Зарядка трехфазным переменным током в соответствии с *ГОСТ Р МЭК 62196-1*, *ГОСТ Р МЭК 62196-2* и *[49]*
DC_core	Зарядка постоянным током в соответствии с *ГОСТ Р МЭК 62196-1*, *ГОСТ Р МЭК 62196-2* и *[49]* через основные контакты
DC_extended	Зарядка постоянным током с использованием дополнительных контактов разъемов конфигурации EE или FF по *[49]*
DC_combo_core	Зарядка постоянным током с использованием основных контактов конфигурации EE или FF по *[49]*
DC_unique	Зарядка постоянным током с использованием специального разъема для постоянного тока

Примечание — SECC может обеспечивать несколько вариантов услуг по зарядке. В зависимости от данных опций EVCC должен выбрать одну из предлагаемых опций. Например, если EVSE предлагает AC_single_phase_core и DC_core, EVCC должен выбрать либо AC_single_phase_core, либо DC_core, потому что обе опции одновременно не могут технически поддерживаться [см. EnergyTransferModeType, F.6 (приложение F)].

8.5.2.5 CertificateChainType

Этот тип данных сохраняет сертификат клиента и все сертификаты в цепочке. Корневой сертификат не входит в этот тип данных. В особом (маловероятном) случае сертификат клиента подписывается непосредственно корнем, поле «SubCertificates» остается пустым. Во всех других случаях это поле содержит запрошенный список сертификатов Sub-CA-certificates для прослеживания пути от сертификата клиента до корня.

[V2G2-274] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 61.

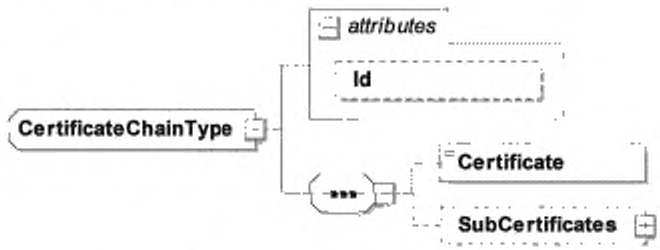


Рисунок 61 — Диаграмма — CertificateChainType

[V2G2-275] Элемент сообщения должен использоваться, как определено в таблице 64.

Таблица 64 — Семантика и определение типа для CertificateChainType

Имя элемента	Тип	Семантика
Id	simpleType: ID цепочка см. определение типа в приложении F	Факультативно: этот атрибут используется для ссылки на тело сообщения в заголовке подписи, когда необходимо применить подпись. Используется только в ответных сообщениях
Certificate	simpleType: certificateType SessionIDType: hexBinary (макс. длина: 800) см. определение типа в F.6 (приложение F)	Сертификат x.509v3 («клиентский» сертификат). Сертификат, кодированный по правилам DER
SubCertificates	complexType: SubCertificatesType см. 8.5.2.26	Факультативно: цепочка со всеми субсертификатами до корневого сертификата (исключая корневой сертификат)

8.5.2.6 MeterInfoType

[V2G2-276] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2. EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 62.

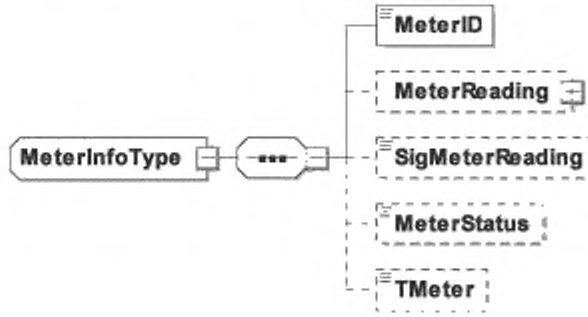


Рисунок 62 — Диаграмма — MeterInfoType

[V2G2-277] Элемент сообщения должен использоваться, как определено в таблице 65.

Таблица 65 — Семантика и определение типа для MeterInfoType

Имя элемента	Тип	Семантика
MeterID	simpleType: meterIDType цепочка (макс. длина: 32) см. определение типа в F.6 (приложение F)	Идентификатор прибора учета EVSE
MeterReading	simpleType: unsignedLong см. определение типа в F.6 (приложение F)	Факультативно: текущее показание прибора учета EVSE в ватт-часах

Окончание таблицы 65

Имя элемента	Тип	Семантика
SigMeterReading	simpleType: sigMeterReadingType SessionIDType: hexBinary (макс. длина: 64) см. определение типа в F.6 (приложение F)	Факультативно: подпись показания прибора учета. Эта подпись генерируется прибором учета EVSE. Она не проверяется EVCC. Она может быть использована системой SA для расчета, если локальные правила учета это разрешают
MeterStatus	simpleType: meterStatusType short см. определение типа в F.6 (приложение F)	Факультативно: текущее состояние прибора учета. Определение содержания MeterStatus не входит в область применения настоящего стандарта. Это содержание может быть определено оператором EVSE или электросетью
TMeter	simpleType: short см. определение типа в F.6 (приложение F)	Факультативно: метка текущего времени прибора учета в формате метки времени системы Unix

[V2G2-830] При обработке элемента сообщений MeterReading SECC и EVCC должны применять базовую единицу измерения ватт-час (Вт·ч).

[V2G2-831] При использовании элемента сообщений MeterReading SECC и EVCC должны применять определение диапазона значения в соответствии с таблицей 66.

Таблица 66 — Определение диапазона значения элемента сообщения MeterReading

Имя элемента	Условное обозначение единицы	Единица физической величины	Минимальное значение	Максимальное значение
MeterReading	Wh	ватт-час	0	999,999,999

8.5.2.7 PhysicalValueType

[V2G2-278] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 63.

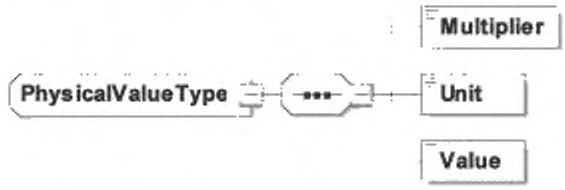


Рисунок 63 — Диаграмма — PhysicalValueType

[V2G2-279] Элемент сообщения должен использоваться, как определено в таблице 67.

Таблица 67 — Семантика и определение типа для PhysicalValueType

Имя элемента	Тип	Семантика
Multiplier	simpleType: unitMultiplierType байт (диапазон: –3...+3) см. определение типа в F.6 (приложение F)	Множитель определяет показатель степени для основания 10 (десять). Окончательная физическая величина определяется выражением: Значение * 10 ^ Множитель [в единицах измерения]

Окончание таблицы 67

Имя элемента	Тип	Семантика
Unit	simpleType: unitSymbolType enumeration см. определение типа в F.6 (приложение F)	Единица измерения физической величины
Value	simpleType: short см. определение типа в F.6 (приложение F)	Значение, которое должно быть умножено

[V2G2-832] Для всех элементов сообщений типа PhysicalValueType SECC и EVCC должны применять диапазон значений и определение единиц измерения в соответствии с таблицей 68.

Таблица 68 — Диапазон значений и определение единиц изменения для элементов сообщений, использующих PhysicalValueType

Имя элемента	Обозначение типа величины	Наименование единицы измерения физической величины	Минимальное значение	Максимальное значение
ChargingProfileEntryMaxPower	W	ватт	0	200 000
Eamount	Wh	ватт-час	0	200 000
EVEnergyCapacity	Wh	ватт-час	0	200 000
EVEnergyRequest	Wh	ватт-час	0	200 000
EVMaxCurrent	A	ампер	0	400
EVMaximumCurrentLimit	A	ампер	0	400
EVMaximumPowerLimit	W	ватт	0	200 000
EVMaximumVoltageLimit	V	вольт	0	1000
EVMaxVoltage	V	вольт	0	1000
EVMinCurrent	A	ампер	0	400
EVSECurrentRegulationTolerance	A	ампер	0	400
EVSEEnergyToBeDelivered	Wh	ватт-час	0	200 000
EVSEMaxCurrent	A	ампер	0	400
EVSEMaximumCurrentLimit	A	ампер	0	400
EVSEMaximumPowerLimit	W	ватт	0	200 000
EVSEMaximumVoltageLimit	V	вольт	0	1000
EVSENominalVoltage	V	вольт	0	1000
EVSEMinimumCurrentLimit	A	ампер	0	400
EVSEMinimumVoltageLimit	V	вольт	0	1000
EVSEPeakCurrentRipple	A	ампер	0	400
EVSEPresentCurrent	A	ампер	0	400
EVSEPresentVoltage	V	вольт	0	1000

Окончание таблицы 68

Имя элемента	Обозначение типа величины	Наименование единицы измерения физической величины	Минимальное значение	Максимальное значение
EVTargetCurrent	A	ампер	0	400
EVTargetVoltage	V	вольт	0	1000
Pmax	W	ватт	0	200 000
RemainingTimeToBulkSoC	s	секунда	0	172800
RemainingTimeToFullSoC	s	секунда	0	172800
startValue	W	ватт	0	200 000

8.5.2.8 NotificationType

[V2G2-280] Этот необязательный элемент сообщения включен в заголовок сообщения V2G и позволяет уведомить принимающее устройство о синтаксической ошибке анализа или любой другой ошибке, связанной с расшифровкой сообщения V2G. Элемент сообщения должен быть реализован в соответствии с рисунком 64.

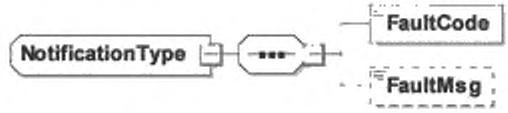


Рисунок 64 — Диаграмма — NotificationType

[V2G2-281] Элемент сообщения должен использоваться, как определено в таблице 69.

Таблица 69 — Семантика и определение типа для NotificationType

Имя элемента	Тип	Семантика
FaultCode	simpleType: faultCodeType enumeration см. определение типа в F.6 (приложение F)	Определяет ошибку синтаксического анализа или какую-либо другую ошибку, относящуюся к декодированию сообщения V2G
FaultMsg	simpleType: цепочка (макс. длина: 64) см. определение типа в F.6 (приложение F)	Факультативно: включает относящуюся к ошибке информацию, которая может быть использована для анализа ошибки

8.5.2.9 PaymentOptionListType

[V2G2-282] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 65.



Рисунок 65 — Диаграмма — PaymentOptionListType

[V2G2-283] Элемент сообщения должен использоваться, как определено в таблице 70.

Таблица 70 — Семантика и определение типа для PaymentOptionListType

Имя элемента	Тип	Семантика
PaymentOption	simpleType: paymentOptionType enumeration см. определение типа в F.6 (приложение F)	Этот тип включает список опций оплаты, который SECC предлагает EVCC, указывая способы, которые могут быть выбраны для оплаты услуг. EVCC может выбрать только один способ оплаты для всех услуг, используемых им

8.5.2.10 ChargingProfileType

[V2G2-284] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 66.

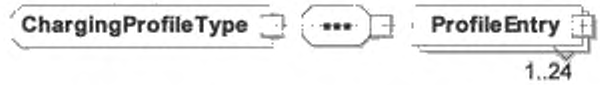


Рисунок 66 — Диаграмма — ChargingProfileType

[V2G2-606] Элемент сообщения должен использоваться, как определено в таблице 71.

Таблица 71 — Семантика и определение типа для ChargingProfileType

Имя элемента	Тип	Семантика
ProfileEntry	complexType: ProfileEntryType см. 8.5.2.11	Элемент, используемый для формирования записи индивидуального профиля зарядки. Число сервисных элементов ProfileEntry ограничивается восемью

8.5.2.11 ProfileEntryType

[V2G2-288] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 67.

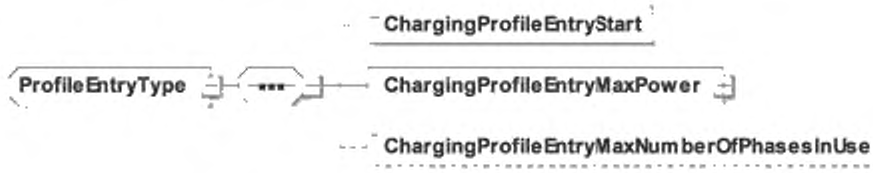


Рисунок 67 — Диаграмма — ProfileEntryType

[V2G2-607] Элемент сообщения должен использоваться, как определено в таблице 72.

Таблица 72 — Семантика и определение типа для ProfileEntryType

Имя элемента	Тип	Семантика
ChargingProfileEntryStart	simpleType: unsignedInt см. определение типа в F.6 (приложение F)	Время, когда chargingProfileEntry начинает действовать. Сдвиг в секундах от настоящего момента времени

Окончание таблицы 72

Имя элемента	Тип	Семантика
ChargingProfileEntryMaxPower	complexType: PhysicalValueType см. 8.5.2.7	Максимальная мощность в ваттах, потребляемая EV в рамках текущей записи профиля зарядки (начиная от ChargingProfileEntryStart)
ChargingProfileEntryMaxNumberOfPhasesInUse	simpleType: byte (мин. значение: 1; макс. значение: 3) см. определение типа в F.6 (приложение F)	Факультативно: этот элемент используется EV для указания максимального количества фаз, которые будут использоваться в течение периода, определенного ChargingProfileEntryStart, этого ProfileEntry

[V2G2-289] Значение элемента ChargingProfileEntryStart должно определяться как точка во времени, когда элемент ProfileEntryType становится активным.

[V2G2-290] Значение элемента ChargingProfileEntryStart в списке элементов ProfileEntryType должно определяться моментом времени, когда элемент ProfileEntryType становится неактивным.

Примечание 1 — **[V2G2-289]** и **[V2G2-290]** определяют период времени, в течение которого элемент ProfileEntryType активен.

[V2G2-291] Последний элемент в списке элементов типа ProfileEntryType активен до обновления списка в соответствии с **[V2G2-305]**.

[V2G2-292] Значение элемента ChargingProfileEntryMaxPower должно определяться как максимальная мощность в ваттах, потребляемая EV в течение активного периода элемента типа ProfileEntryType.

[V2G2-293] Значения элемента ChargingProfileEntryMaxPower должны быть равны или меньше, чем предельные значения в соответствующих элементах PMaxScheduleType (см. 8.5.2.14), которые даны в сообщении ChargeParameterDiscoveryRes.

[V2G2-829] Если EV может определить количество фаз, которое будет использоваться в течение отдельного интервала времени (ChargingProfileEntryStart), EVCC должен указать для каждого интервала времени максимальное количество фаз, которое он намерен использовать в данный интервал времени, путем использования параметра ChargingProfileEntryMaxNumberOfPhasesInUse.

Примечание 2 — Информация об использовании фаз в определенный интервал времени может быть учтена SECC для оптимизации распределения энергии по имеющимся фазам.

8.5.2.12 SAScheduleListType

[V2G2-294] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 68.



Рисунок 68 — Диаграмма — SAScheduleListType

[V2G2-608] Элемент сообщения должен использоваться, как определено в таблице 73.

Таблица 73 — Семантика и определение типа для SAScheduleListType

Имя элемента	Тип	Семантика
SAScheduleTuple	complexType: SAScheduleTupleType см. 8.5.2.13	Включает несколько кортежей графиков от SA. Число сервисных элементов SAScheduleTuple ограничивается тремя

[V2G2-296] EVCC может реализовать механизм для сравнения различных элементов SAScheduleTuple с целью оптимизации графика зарядки с учетом любой конкретной стоимости в соответствии с **[V2G2- 803]** и **[V2G2-337]**.

[V2G2-297] Первый элемент SAScheduleTuple в SAScheduleListType должен определяться как SASchedule по умолчанию.

[V2G2-298] Если EVCC не способен сравнивать различные элементы SAScheduleTuple или сравнение не удается, EVCC должен выбрать SAScheduleTuple в соответствии с **[V2G2-297]**.

8.5.2.13 SAScheduleTupleType

[V2G2-299] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 69.

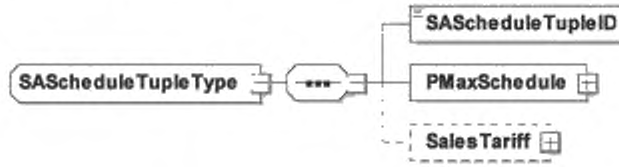


Рисунок 69 — Диаграмма — SAScheduleTupleType

[V2G2-609] Элемент сообщения должен использоваться, как определено в таблице 74.

Таблица 74 — Семантика и определение типа для SAScheduleTupleType

Имя элемента	Тип	Семантика
SAScheduleTupleID	simpleType: SAIDType unsignedByte см. определение типа в F.6 (приложение F)	Уникальный идентификатор для элемента SAScheduleTuple в течение сеанса зарядки. Идентификатор SA остается уникальным идентификатором для одного графика в течение сеанса зарядки
PMaxSchedule	complexType: PMaxScheduleType см. 8.5.2.14	Инкапсулирующий элемент, описывающий все релевантные детали для одного PMaxSchedule от SA
SalesTariff	complexType: SalesTariffType см. 8.5.2.16	Факультативно: инкапсулирующий элемент, описывающий все релевантные детали для одного SalesTariff от SA

[V2G2-773] SECC должен использовать значения 1—255 для параметра SAScheduleTupleID.

[V2G2-300] Элемент SAScheduleTupleID должен быть уникальным среди всех элементов SAScheduleTuple в SAScheduleListType и уникально идентифицировать кортеж элементов PMaxSchedule и SalesTariff во время всего сеанса зарядки.

[V2G2-301] SECC должен предоставлять элемент PMaxSchedule на основе предельных параметров локальной инфраструктуры, если ни один из SA не предоставляет график электросети.

[V2G2-303] Сумма индивидуальных интервалов времени, описанных в PMaxSchedule (см. 8.5.2.14) и SalesTariff (см. 8.5.2.16), которая дана в сообщении ChargeParameterDiscoveryRes, должна соответствовать периоду времени, указанному EVCC в элементе DepartureTime сообщения ChargeParameterDiscoveryReq.

[V2G2-304] Если EVCC не дал целевую настройку DepartureTime (см. 8.4.3.8.2 и 8.5.3.2) в сообщении ChargeParameterDiscoveryReq, сумма индивидуальных интервалов времени, описанных в PMaxSchedule и SalesTariff, которые даны в сообщении ChargeParameterDiscoveryRes, должна быть больше или равна 24 ч.

[V2G2-305] Если число элементов SalesTariffEntry в SalesTariff или число элементов PMaxScheduleEntry в PMaxSchedule, переданных SA, не покрывает весь период времени до DepartureTime, целевая настройка EAmount (см. 8.4.3.8.2 и 8.5.3.2) не была выполнена и сеанс связи не был закончен, то как только последний элемент SalesTariffEntry или последний элемент PMaxScheduleEntry станет активным, EVCC должен запросить новый элемент типа SAScheduleListType путем отправки нового сообщения ChargeParameterDiscoveryReq.

Примечание 1 — EVCC может запросить обновление тарифной информации, применив процесс пересогласования в любое время цикла зарядки.

[V2G2-306] Если количество элементов SalesTariffEntry, переданных SA, не покрывает весь период времени до DepartureTime, EVCC должен оптимизировать график на базе имеющейся информации.

Примечание 2 — Алгоритм оптимизации профиля зарядки не рассматривается в настоящем стандарте.

[V2G2-307] В случае PnC и если Tariff Table используется SA, последний должен подписать поле SalesTariff типа SalesTariffType. В случае EIM это поле может подписать SA.

[V2G2-905] В случае, если SA использует механизм тарифных ставок, SECC должен направить эту информацию EVCC, используя поле SalesTariff типа SalesTariffType.

Примечание 3 — Если SA неизвестно, какой режим аутентификации используется во время связи EVCC-SECC (EIM/PnC), он может всегда подписывать SalesTariff.

[V2G2-308] SECC не должен изменять подпись сообщения, связанную с элементом SalesTariff, при получении тарифной информации от SA.

Примечание 4 — Предполагается, что структура данных, используемая SA для выдачи SalesTariff, идентична структуре данных, описанной в настоящем стандарте.

[V2G2-309] SECC должен «скопировать» значение подписи, полученное от SA, и передать это значение в заголовке сообщения ChargeParameterDiscoveryRes (см. 8.4.3.8.3 и 8.3.3).

[V2G2-906] Если элемент SalesTariff подписывается, то он должен подписываться тем же самым закрытым ключом, который был использован для выдачи листового сертификата контракта, который EVCC использовал во время соединения для аутентификации контракта (PnC).

Примечание 5 — Это позволяет EVCC проверить надежность таблицы тарифных ставок, выполнив только одну проверку на базе ECC-криптографии. В реализации SECC/SA должна использоваться цепочка сертификата контракта EVCC, которая была передана в элементе ContractSignatureCertChain сообщения PaymentDetailsReq во время аутентификации контракта для идентификации закрытого ключа, который должен использоваться для подписи таблицы тарифных ставок. Это требование может быть выполнено путем взаимодействия между SECC и SA, включая онлайновую связь, или подписанные таблицы тарифных ставок могут быть помещены в кэш.

[V2G2-907] В случае PnC после получения поля SalesTariff типа SalesTariffType EVCC должен проверить подпись с помощью того же сертификата Sub-CA, который был использован для выдачи сертификата контракта, который EVCC ранее использовал для аутентификации контракта (PnC). Если эта проверка не удается, EVCC должен рассматривать SalesTariff как недействительный. См. **[V2G2-908]**.

Примечание 6 — В случае EIM EVCC может игнорировать подпись (если она существует).

Примечание 7 — EVCC требуется выполнить только одну проверку на базе ECC-криптографии, потому что подпись таблицы тарифных ставок была создана тем же закрытым ключом, использованным для выдачи листового сертификата контракта, который EVCC применил во время цикла аутентификации контракта (PnC). EVCC, таким образом, уже имеет хранящийся сертификат для проверки таблицы тарифных ставок (как часть цепочки сертификата контракта).

[V2G2-908] Если EVCC рассматривает SalesTariff как недействительный, он должен игнорировать таблицу тарифных ставок, т. е. поведение EVCC должно быть таким же, как если бы таблицы тарифных ставок не были получены. Более того, EVCC может разорвать соединение, а затем его восстановить.

8.5.2.14 PMaxScheduleType

[V2G2-310] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 70.



Рисунок 70 — Диаграмма — PMaxScheduleType

[V2G2-610] Элемент сообщения должен использоваться, как определено в таблице 75.

Таблица 75 — Семантика и определение типа для PMaxScheduleType

Имя элемента	Тип	Семантика
PMaxScheduleEntry	complexType: PMaxScheduleEntryType см. 8.5.2.15	Список элементов PMaxScheduleEntry. Число элементов PMaxScheduleEntry ограничивается параметром MaxEntriesSAScheduleTuple (в соответствии с **[V2G2-261]**). Максимальное значение должно быть 1024

8.5.2.15 PMaxScheduleEntryType

[V2G2-313] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 71.

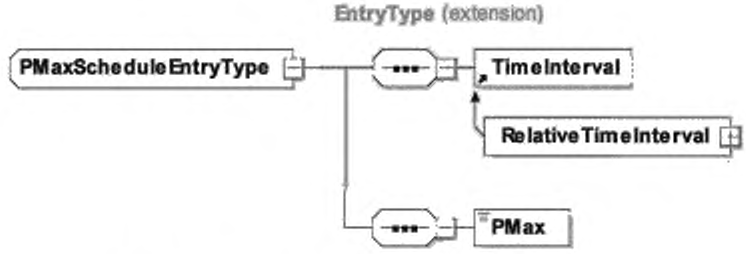


Рисунок 71 — Диаграмма — PMaxScheduleEntryType

[V2G2-611] Элемент сообщения должен использоваться, как определено в таблице 76.

Таблица 76 — Семантика и определение типа для PMaxScheduleEntryType

Имя элемента	Тип	Семантика
RelativeTimeInterval	complexType: RelativeTimeInterval заменяет абстрактный элемент TimeInterval см. 8.5.2.18	Расширяет TimeIntervalType и определяет интервал времени, в течение которого действительна запись PMaxScheduleEntry, на базе относительного времени
PMax	complexType: PhysicalValueType см. 8.5.2.7	Определяет максимальное значение мощности за то время, пока EV подключен к EVSE. Это значение представляет собой суммарную мощность всех фаз

[V2G2-314] Расширение элемента TimeInterval в соответствии с **[V2G2-330]** должно определять активный период времени для соответствующего исходного элемента типа PMaxScheduleEntryType.

[V2G2-315] Элемент PMax должен определять максимальное значение мощности за то время, пока EV подключен к EVSE, при условии, что элемент типа PMaxScheduleEntryType активен.

Примечание — В случае, если EV переходит с трехфазной зарядки на однофазную, EV должен пересчитать PMax.

8.5.2.16 SalesTariffType

Таблица тарифных ставок, приведенная в настоящем стандарте, обеспечивает средства оптимизации графика зарядки в EVCC на базе информации о стоимости. Такая информация предоставляется в составе тарифных данных. Термин «стоимость» относится к любому виду стоимости, которая описана в настоящем стандарте. Однако в настоящем стандарте не предоставлена абсолютная ценовая информация или рекомендации по тарифам для пользователя EV. Эта функциональность может быть реализована с помощью других средств, не рассмотренных в настоящем стандарте.

[V2G2-316] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 72.

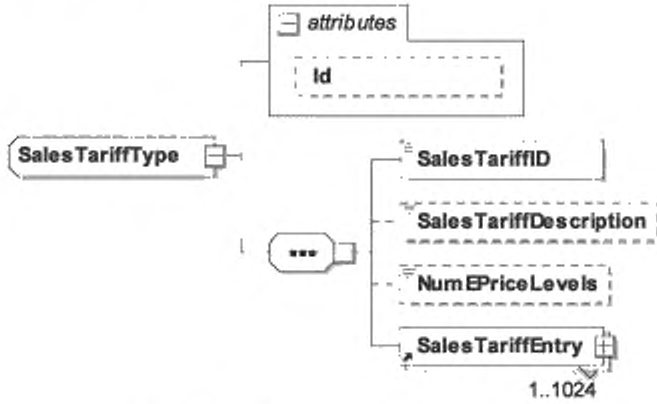


Рисунок 72 — Диаграмма — SalesTariffType

[V2G2-612] Элемент сообщения должен использоваться, как определено в таблице 77.

Таблица 77 — Семантика и определение типа для SalesTariffType

Имя элемента	Тип	Семантика
Id	simpleType: ID цепочка см. определение типа в F.6 (приложение F)	Факультативно: этот атрибут используется для ссылки на элемент сообщения в заголовке подписи, когда ее необходимо применить
SalesTariffID	simpleType: SAIDType short см. определение типа в F.6 (приложение F)	Идентификатор SalesTariff, используемый для идентификации одного тарифа. Идентификатор SA остается уникальным идентификатором для одного графика в течение сеанса зарядки
SalesTariffDescription	simpleType: tariffDescriptionType цепочка (макс. длина: 32) см. определение типа в F.6 (приложение F)	Факультативно: удобочитаемое название/короткое описание тарифной ставки, например, для HMI дисплея
NumEPriceLevels	simpleType: unsignedByte см. определение типа в F.6 (приложение F)	Факультативно: определяет общее число отдельных уровней стоимости, используемых для всех предусмотренных элементов SalesTariff

Окончание таблицы 77

Имя элемента	Тип	Семантика
SalesTariffEntry	complexType: SalesTariffEntryType см. 8.5.2.17	Инкапсулирующий элемент, описывающий все релевантные детали для одного временного интервала действия тарифа SalesTariff. Число элементов SalesTariffEntry ограничивается параметром MaxEntriesSAScheduleTuple (в соответствии с **[V2G2-261]**). Максимальное значение должно быть 1024

[V2G2-317] Значение элемента SalesTariffID должно идентифицировать элемент типа SalesTariffType в течение всего сеанса зарядки.

[V2G2-318] Элемент NumEPriceLevels должен быть определен как общее число отдельных элементов EPriceLevel, используемых для всех элементов SalesTariff.

Примечание 1 — Если используется несколько тарифных ставок, они должны исходить от одного SA. Поэтому одна подпись такого SA применяется ко всем тарифным ставкам.

Пример 1 — Если в составе элемента SAScheduleList дается только одна тарифная ставка SalesTariff с разным уровнем цен, установленными, например, для интервалов максимальной, средней, минимальной нагрузки электросети, элемент NumEPriceLevels должен быть установлен в результирующее значение 3.

Пример 2 — Если в составе элемента SAScheduleList дается только одна тарифная ставка SalesTariff с постоянным уровнем цены для всех периодов времени, элемент NumEPriceLevels должен быть установлен в результирующее значение 1.

Пример 3 — Если в составе элемента SAScheduleList даются две тарифные ставки, при этом первая ставка SalesTariff T1 имеет три уровня цен, а вторая SalesTariff T2 имеет два уровня цен и все уровни цен обеих ставок T1 и T2 отличаются друг от друга, элемент NumEPriceLevels должен быть установлен в результирующее значение 5.

Пример 4 — Если в составе элемента SAScheduleList даются две тарифные ставки, при этом первая ставка SalesTariff T1 имеет три уровня цен, вторая ставка SalesTariff T2 имеет два уровня цен и один уровень цены из каждой ставки T1 равен одному уровню цены ставки T2, элемент NumEPriceLevels должен быть установлен в результирующее значение 4.

Примечание 2 — См. подробные примеры в J.3.7 (приложение J).

[V2G2-805] Если ценовая информация отсутствует, но элемент типа SalesTariffType присутствует для обеспечения возможности передачи информации о стоимости потребления (только CarbonDioxideEmission и RenewableGenerationPercentage), элементы NumEPriceLevels и EPriceLevel не должны включаться в сообщение ChargeParameterDiscoveryRes.

8.5.2.17 SalesTariffEntryType

[V2G2-321] В зависимости от выбранного(ых) набора(ов) сообщений, описаного(ых) в 8.6.2, SECC и EVCC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 73.

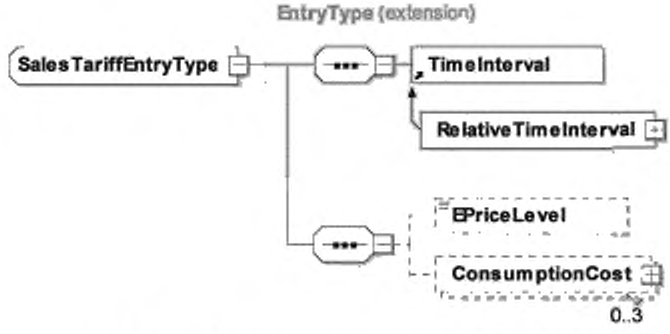


Рисунок 73 — Диаграмма — SalesTariffEntryType

[V2G2-613] Элемент сообщения должен использоваться, как определено в таблице 78.

Таблица 78 — Семантика и определение типа для SalesTariffEntryType

Имя элемента	Тип	Семантика
RelativeTimeInterval	complexType: RelativeTimeInterval заменяет абстрактный элемент TimeInterval см. 8.5.2.18	Расширяет TimeIntervalType и определяет временной интервал, для которого SalesTariffEntry действителен на базе относительного времени
EPriceLevel	simpleType: unsignedByte см. определение типа в F.6 (приложение F)	Факультативно: определяет уровень цены данного SalesTariff Entry (по отношению к NummEPriceLevels). Меньшие значения для EPriceLevel соответствуют более дешевому TariffEntry. Большие значения для EPriceLevel соответствуют более дорогому TariffEntry
ConsumptionCost	complexType: ConsumptionCostType см. 8.5.2.19	Факультативно: определяет дополнительные значения для прочей информации о ценах и/или альтернативных затратах
Примечание — Определение EPriceLevel делает возможной простую оптимизацию на базе цены EVCC без какого-либо элемента ConsumptionCost.		

[V2G2-322] Любое расширение элемента TimeInterval должно определять активный период времени для соответствующего исходного элемента типа SalesTariffEntryType.

[V2G2-803] Значение элемента EPriceLevel должно указывать на ценовой уровень, который используется для интервала данного SalesTariffEntry.

Примечание 1 — Значение элемента EPriceLevel определяет, какой ценовой уровень действует для данного интервала SalesTariff. Значение элемента EPriceLevel не указывает фактическую стоимость в пределах интервала действия данного SalesTariffEntry. Применимые относительные стоимости могут быть факультативно определены в элементе ConsumptionCost, который описан в 8.5.2.19.

Примечание 2 — Элемент EPriceLevel может быть использован EV в дополнение к предельным параметрам мощности для расчета оптимизированного по цене графика зарядки.

[V2G2-324] Действительный диапазон для значения элемента EPriceLevel должен быть установлен от 0 до значения элемента NumEPriceLevels (см. 8.5.2.16), включая граничные значения.

[V2G2-802] Если значение NumEPriceLevels равно «0» (и только тогда), значение EPriceLevel должно быть «0».

Примечание 3 — Элементы EPriceLevel и NumEPriceLevels устанавливаются на «0», если SA хочет передавать стоимость потребления только для типа RenewableGenerationPercentage и CarbonDioxideEmission.

[V2G2-325] EPriceLevel должен соответствовать следующему правилу: чем меньше значения EPriceLevel, тем дешевле фактический ценовой уровень в соответствующем интервале. Чем больше значения EPriceLevel, тем дороже фактический ценовой уровень в соответствующем интервале.

8.5.2.18 RelativeTimeIntervalType

[V2G2-327] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 74.

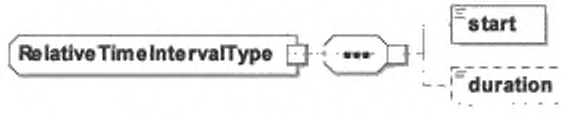


Рисунок 74 — Диаграмма — RelativeTimeIntervalType

[V2G2-614] Элемент сообщения должен использоваться, как определено в таблице 79.

Таблица 79 — Семантика и определение типа для RelativeTimeIntervalType

Имя элемента	Тип	Семантика
duration	simpleType: unsignedInt см. определение типа в F.6 (приложение F)	Факультативно: продолжительность интервала в секундах
start	simpleType: unsignedInt см. определение типа в F.6 (приложение F)	Начало интервала в секундах от NOW

[V2G2-328] Значение элемента начала должно быть установлено в секундах от текущего момента времени NOW.

[V2G2-329] Значение элемента начала должно одновременно определять время начала данного интервала и время окончания предыдущего интервала.

[V2G2-330] Значение элемента продолжительности должно быть установлено как период времени в секундах.

[V2G2-331] Элемент продолжительности должен использоваться только для последнего интервала действия SalesTariff (см. 8.5.2.16) или PMaxSchedule (см. 8.5.2.14).

Примечание 1 — Данный элемент указывает на окончание времени действия предоставленной информации о SalesTariff или PMaxSchedule.

Примечание 2 — Определение относительного времени используется для исключения необходимости синхронизации часов. Поскольку время двустороннего обмена сообщениями может занимать несколько секунд, SECC должен считать приемлемым небольшой льготный период. Например, если EVCC начинает зарядку раньше или завершает позднее, чем задано в элементе начала, SECC не должен заканчивать сеанс зарядки с информацией об ошибке. Он должен принимать это событие в пределах льготного периода времени как корректное.

[V2G2-833] В момент, когда начинается новый временной интервал ChargingProfile, как указывает ChargingProfileEntryStart (n+1), EVSE должно допустить потребление мощности EV, как указано в ChargingProfileEntryMaxPower (n) для предыдущего временного интервала, который определен ChargingProfileEntryStart (n), со временем до 120 с дольше, чем определено в ChargingProfileEntryStart (n+1), даже если значение ChargingProfileEntryMaxPower (n) выше, чем ChargingProfileEntryMaxPower (n+1).

[V2G2-834] В момент, когда начинается новый временной интервал ChargingProfile, как указывает ChargingProfileEntryStart (n+1), EVSE должно допустить потребление мощности EV, как указано в ChargingProfileEntryMaxPower (n+1) для предыдущего временного интервала, который определен ChargingProfileEntryStart (n), на время до 120 с раньше, чем определено в ChargingProfileEntryStart (n+1), даже если значение ChargingProfileEntryMaxPower (n+1) выше, чем ChargingProfileEntryMaxPower.

8.5.2.19 ConsumptionCostType

[V2G2-332] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 75.

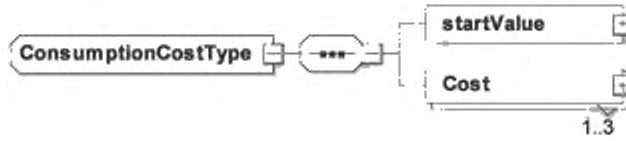


Рисунок 75 — Диаграмма — ConsumptionCostType

[V2G2-615] Элемент сообщения должен использоваться, как определено в таблице 80.

Таблица 80 — Семантика и определение типа для ConsumptionCostType

Имя элемента	Тип	Семантика
Cost	complexType: CostType см. 8.5.2.20	Инкапсулирующий элемент, описывающий все релевантные детали стоимости для данного блока потребления в данном TariffEntry
startValue	complexType: PhysicalValueType см. определение типа в F.6 (приложение F)	Самый низкий уровень потребления, который определяет начальную точку данного блока потребления. Интервал блока продолжается до начала следующего интервала

[V2G2-328] Первый элемент в списке элементов типа ConsumptionCostType должен всегда начинаться со startValue, установленного в значение 0.

[V2G2-334] Максимальное число элементов Cost в ConsumptionCostType должно ограничиваться числом различных типов перечисления, заданным в CostKindType (см. таблицу 82).

8.5.2.20 CostType

[V2G2-335] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 76.

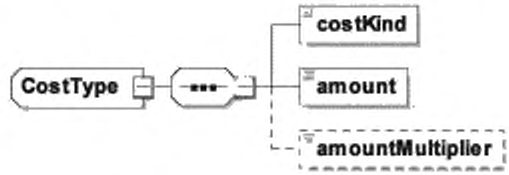


Рисунок 76 — Диаграмма — CostType

[V2G2-616] Элемент сообщения должен использоваться, как определено в таблице 81 и таблице 82.

Таблица 81 — Семантика и определение типа для CostType

Имя элемента	Тип	Семантика
amount	simpleType: unsignedInt см. определение типа в F.6 (приложение F)	Оценочная или фактическая стоимость за кВт·ч
amountMultiplier	simpleType: unitMultiplierType (мин. значение: –3; макс. значение: 3), см. F.6 (приложение F)	Факультативно: amountMultiplier определяет показатель степени основания 10 (десять). Окончательное значение определяется выражением: amount * 10 ^ amountMultiplier
costKind	simpleType: CostKindType enumeration см. определение типа в F.6 (приложение F)	Вид стоимости, указанный в значении элемента сообщения (см. в таблице 82 описание семантики отдельных перечисляемых значений, заданных для этого типа)

Таблица 82 — Семантика для CostKindType

CostKindType	Семантика
relativePricePercentage	Относительное значение. Цена за кВт·ч как процентная доля от максимальной цены в любом из тарифов, указанных для EV
RenewableGenerationPercentage	Относительное значение. Процентная доля выработки (электроэнергии) из возобновляемых источников от общей выработки
CarbonDioxideEmission	Абсолютное значение. Выбросы двуокиси углерода в граммах на кВт·ч

[V2G2-336] Единицей измерения любых затрат выступают кВт·ч.

[V2G2-337] Применяемые затраты на единицу измерения согласно **[V2G2-336]** должны быть определены финальной единицей оплаты.

[V2G2-338] В пределах одного типа CostType такая единица оплаты должна ссылаться на соответствующий элемент costKind.

[V2G2-339] Для относительных видов стоимости (см. таблицу 82) единица оплаты должна быть определена как процентная доля от наибольшей стоимости данного вида.

[V2G2-775] Единица оплаты в случае расчета в относительных единицах не должна превышать 100 %.

[V2G2-340] В случае расчета в абсолютных величинах единица оплаты выражается в конечной стоимости.

Пример — Если множитель равен 2 и переданное значение равно 1, то рассчитываемое значение равно 1 * 10 ^ 2 = 100.

[V2G2-806] Если CostKindType в одном SalesTariff включает тип абсолютной стоимости, все другие тарифные ставки SalesTariffs, включенные в сообщение ChargeParameterDiscoveryRes, также должны включать данный тип абсолютной стоимости для обеспечения возможности сравнения разных тарифных ставок.

[V2G2-807] Если CostKindType в одном SalesTariff включает тип относительной стоимости, все другие тарифные ставки, включенные в сообщение ChargeParameterDiscoveryRes, также должны включать данный тип относительной стоимости для обеспечения возможности сравнения разных тарифных ставок.

[V2G2-808] Если тип относительной стоимости используется в SalesTariff, он должен быть приведен к общей базе по всем элементам SalesTariff поставщиком данных о SalesTariff.

Примечание — Если это условие не выполняется, EVCC не способен сравнить разные типы тарифов.

[V2G2-772] Если хотя бы одна запись SalesTarifEntry включает элемент ConsumptionCost, все записи SalesTariffEntries, включенные во все SAScheduleTuples, переданные от SECC к EVCC в течение одного конкретного сеанса зарядки, должны включать элементы ConsumptionCost с идентичной структурой (т. е. число CostType и интервалов на SalesTariffEntry должно быть идентичным).

8.5.2.21 ServiceParameterListType

[V2G2-343] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 77.



Рисунок 77 — Диаграмма — ServiceParameterListType

[V2G2-344] Элемент сообщения должен использоваться, как определено в таблице 83.

Таблица 83 — Семантика и определение типа для ServiceParameterListType

Имя элемента	Тип	Семантика
ParameterSet	complexType: ParameterSetType см. 8.5.2.22	Определяет параметры для конкретного идентификатора сервиса serviceID, полученного от SECC в сообщении ServiceDiscoveryRes

8.5.2.22 ParameterSetType

[V2G2-345] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 78.

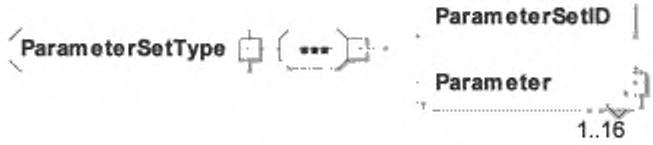


Рисунок 78 — Диаграмма — ParameterSetType

[V2G2-346] Элемент сообщения должен использоваться, как определено в таблице 84.

Таблица 84 — Семантика и определение типа для ParameterSetType

Имя элемента	Тип	Семантика
ParameterSetID	simpleType: short см. определение типа в F.6 (приложение F)	Этот элемент используется для выбора конкретного набора параметров для конкретного идентификатора serviceID сервиса при выборе сервиса с использованием сообщения PaymentServiceSelectionReq
Parameter	complexType: ParameterType см. 8.5.2.23	Этот элемент используется SECC для указания, какие специфичные для сервиса параметры могут быть выбраны для определенного сервиса с использованием ParameterSetID. Число элементов Parameter ограничивается 16

8.5.2.23 ParameterType

[V2G2-347] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 79.

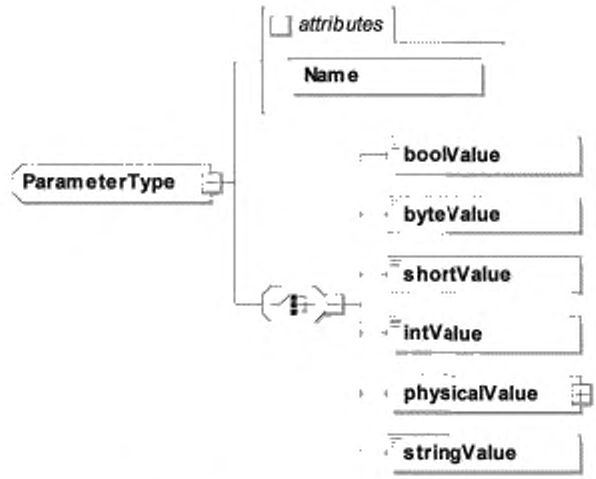


Рисунок 79 — Диаграмма — ParameterType

[V2G2-348] Элемент сообщения должен использоваться, как определено в таблице 85.

Таблица 85 — Семантика и определение типа для ParameterType

Имя элемента	Тип	Семантика
Name (Имя)	simpleType: строка см. определение типа в F.6 (приложение F)	Этот элемент используется для указания имени параметра
Один из элементов: - boolValue - byteValue - shortValue - intValue - physicalValue - stringValue	simpleType: boolean (boolValue) byte (byteValue) short (shortValue) int (intValue) строка (stringValue) см. определение типа в F.6 (приложение F) complexType: PhysicalValueType (physicalValue) см. 8.5.2.7	Этот элемент используется для указания значения параметра, обозначенного именем элемента. Выбор из шести разных типов элементов. Может быть выбран только один для каждого параметра

8.5.2.24 SelectedServiceListType

[V2G2-349] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 80.



Рисунок 80 — Диаграмма — SelectedServiceListType

[V2G2-350] Элемент сообщения должен использоваться, как определено в таблице 86.

Таблица 86 — Семантика и определение типа для SelectedServiceType

Имя элемента	Тип	Семантика
SelectedService	complexType: SelectedServiceType см. 8.5.2.25	Этот элемент используется для указания выбранного ServiceID и соответствующего parameterSet

8.5.2.25 SelectedServiceType

[V2G2-351] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 81.

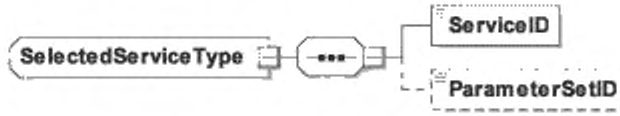


Рисунок 81 — Диаграмма — SelectedServiceType

[V2G2-352] Элемент сообщения должен использоваться, как определено в таблице 87.

Таблица 87 — Семантика и определение типа для SelectedServiceType

Имя элемента	Тип	Семантика
ServiceID	simpleType: unsignedShort см. определение типа в F.6 (приложение F)	Уникальный идентификатор сервиса
ParameterSetID	simpleType: short см. определение типа в F.6 (приложение F)	Этот элемент используется для выбора конкретного набора параметров для конкретного идентификатора ServiceID сервиса при выборе сервиса с использованием сообщения PaymentServiceSelectionReq

8.5.2.26 SubCertificatesType

[V2G2-353] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 82.

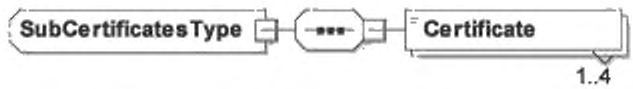


Рисунок 82 — Диаграмма — SubCertificatesType

Следует учитывать, что вследствие **[V2G2-656]** и **[V2G2-009]** элемент типа SubCertificatesType может содержать максимум два элемента типа certificateType.

[V2G2-354] Элемент сообщения должен использоваться, как определено в таблице 88.

Таблица 88 — Семантика и определение типа для SubCertificatesType

Имя элемента	Тип	Семантика
Certificate	simpleType: certificateType base64Binary (макс. длина: 800) см. определение типа в F.6 (приложение F)	Сертификат x.509v3, кодированный по правилам DER

[V2G2-656] Число сертификатов в SubCertificates не должно превышать 2.

8.5.2.27 ListOfRootCertificateIDsType

[V2G2-355] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 83.

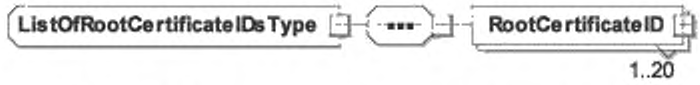


Рисунок 83 — Диаграмма — ListOfRootCertificateIDsType

[V2G2-356] Элемент сообщения должен использоваться, как определено в таблице 89.

Таблица 89 — Семантика и определение типа для ListOfRootCertificateIDsType

Имя элемента	Тип	Семантика
RootCertificateID	simpleType: xmlsig:X509IssuerSerialType строка (макс. длина: 40) см. определение типа в F.6 (приложение F)	Этот элемент сообщения уникальным образом идентифицирует корневой сертификат V2G, установленный в EVCC

8.5.2.28 ContractSignatureEncryptedPrivateKeyType

[V2G2-778] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 84.



Рисунок 84 — Диаграмма — ContractSignatureEncryptedPrivateKeyType

[V2G2-779] Элемент сообщения должен использоваться, как определено в таблице 90.

Таблица 90 — Семантика и определение типа для ContractSignatureEncryptedPrivateKeyType

Имя элемента	Тип	Семантика
Id	simpleType: ID строка см. определение типа в F.6 (приложение F)	Этот атрибут используется для ссылки на элемент сообщения ContractSignatureEncrypted PrivateKey в заголовке подписи, когда подпись необходимо применять

8.5.2.29 DiffieHellmanPublickeyType

[V2G2-780] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 85.

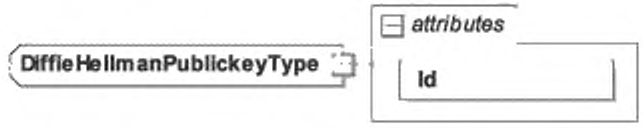


Рисунок 85 — Диаграмма — DiffieHellmanPublickeyType

[V2G2-781] Элемент сообщения должен использоваться, как определено в таблице 91.

Таблица 91 — Семантика и определение типа для DiffieHellmanPublickeyType

Имя элемента	Тип	Семантика
Id	simpleType: ID строка см. определение типа в F.6 (приложение F)	Этот атрибут используется для ссылки на элемент DHpublickey сообщения в заголовке сообщения, когда необходимо применять подпись

8.5.2.30 EMAIDType

[V2G2-782] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 86.

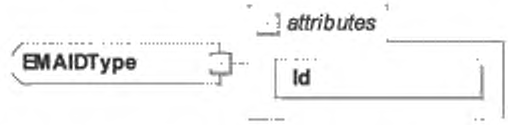


Рисунок 86 — Диаграмма — EMAIDType

[V2G2-783] Элемент сообщения должен использоваться, как определено в таблице 92.

Таблица 92 — Семантика и определение типа для EMAIDType

Имя элемента	Тип	Семантика
Id	simpleType: ID строка см. определение типа в F.6 (приложение F)	Этот элемент используется для ссылки на элемент EMAID сообщения в заголовке сообщения, когда необходимо применять подпись

8.5.3 AC

8.5.3.1 AC_EVSEStatusType

[V2G2-358] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 87.

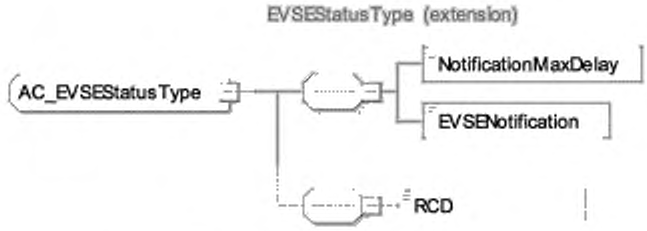


Рисунок 87 — Диаграмма — AC_EVSEStatusType

[V2G2-359] Элемент сообщения должен использоваться, как определено в таблице 93.

Таблица 93 — Семантика и определение типа для AC_EVSEStatusType

Имя элемента	Тип	Семантика
RCD	simpleType: boolean см. определение типа в F.6 (приложение F)	Указывает статус устройства защитного отключения (Residual Current Device — RCD). Если RCD равно True, RCD обнаружило ошибку. Если RCD равно False, RCD не обнаружило ошибку. Этот флаг статуса служит только для информации
NotificationMaxDelay	simpleType: unsignedShort этот элемент унаследован от EVStatusType см. определение типа в F.6 (приложение F)	Эта величина означает время в секундах от момента отправления данного сообщения (относительное время) до непосредственно момента, когда ожидается выполнение действия. SECC использует элемент NotificationMaxDelay в EVSEStatus для указания на время, до наступления которого он ожидает реакции EVCC на запрос действия, указанного в EVSENotification
EVSENotification	simpleType: EVSENotificationType Enumeration этот элемент унаследован от EVStatusType см. определение типа в F.6 (приложение F)	Это значение используется SECC для постановки задачи для EVCC. EVSENotification содержит действие, выполнение которого SECC ожидает от EVCC. Предполагается, что запрошенное действие будет выполнено EVCC до наступления момента времени, указанного в NotificationMaxDelay. Если не подразумевается, что это отложенная временная цель, то ожидается, что EVCC выполнит действие немедленно. Во время нормальной работы значение EVSENotification установлено в значение «поле» (не указано)

Примечание — Поведение EVSE и EV после аварийного отключения электроэнергии (время отключения не равно нулю) не рассматривается в настоящем стандарте. Оно в значительной степени зависит от конкретной реализации системы.

Пример — Реализация продолжения сеанса зарядки после случившегося аварийного отключения электроэнергии может иметь три попытки восстановления, прежде чем SECC перейдет в режим неисправности или ожидания ремонта.

8.5.3.2 AC_EVChargeParameterType

[V2G2-360] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 88.

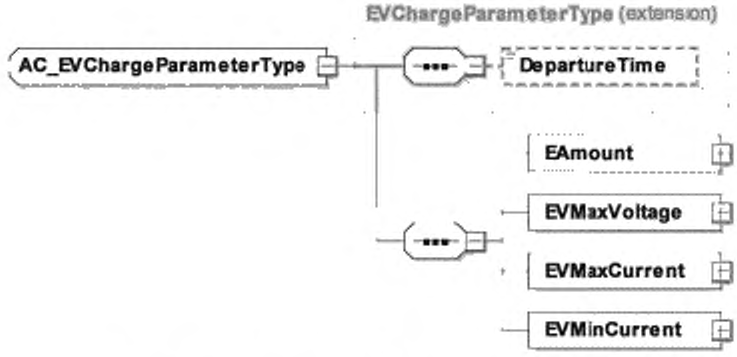


Рисунок 88 — Диаграмма — AC_EVChargeParameterType

[V2G2-361] Элемент сообщения должен использоваться, как определено в таблице 94.

Таблица 94 — Семантика и определение типа для AC_EVChargeParameterType

Имя элемента	Тип	Семантика
DepartureTime	simpleType: unsignedInt этот элемент унаследован от EVChargeParameterType см. определение типа в F.6 (приложение F)	Факультативно: этот элемент используется для указания на время, когда EV намерен завершить процесс зарядки. Представляет собой временной интервал в секундах от момента отправки данного сообщения
EAmount	complexType: PhysicalValueType см. 8.5.2.7	Количество энергии, необходимой для выполнения настроенной пользователем величины в текущем сеансе зарядки. Может содержать требуемый объем электроэнергии и для других целей, отличных от зарядки высоковольтной батареи EV
EVMaxVoltage	complexType: PhysicalValueType см. 8.5.2.7	Среднеквадратичное значение максимально допустимого напряжения для конкретного EV, измеренное между одной фазой и нейтралью
EVMaxCurrent	complexType: PhysicalValueType см. 8.5.2.7	Максимальный ток на фазу, поддерживаемый EV
EVMinCurrent	complexType: PhysicalValueType см. 8.5.2.7	EVMinCurrent используется для того, чтобы сообщить SECC, что зарядка током со значением ниже данного минимума не эффективна для данного транспортного средства с точки зрения соотношения энергия/стоимость. Рекомендуется, чтобы SECC рассматривал это значение во время процесса установки целей (например, таблица тарифных ставок должна учитываться применительно к этому значению). Однако если имеются физические ограничения или ограничения, задаваемые PWM-сигналом, эти ограничения имеют преимущественное значение перед сообщением EVMinCurrent, которое отправляется транспортным средством. Это зависит от реализации, выбирает ли транспортное средство отказ от зарядки с точки зрения эффективности, в случае если EVMinCurrent выше, чем физические ограничения

8.5.3.3 AC_EVSEChargeParameterType

[V2G2-362] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 89.

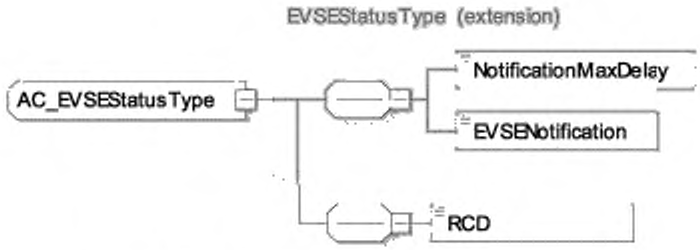


Рисунок 89 — Диаграмма — AC_EVSEChargeParameterType

[V2G2-363] Элемент сообщения должен использоваться, как определено в таблице 95.

Таблица 95 — Семантика и определение типа для AC_EVSEChargeParameterType

Имя элемента	Тип	Семантика
AC_EVSEStatus	complexType: AC_EVSEStatusType см. 8.5.3.1	Актуальное значение тока зарядки EVSE
EVS.ENominalVoltage	complexType: PhysicalValueType см. 8.5.2.7	Напряжение сети, поддерживаемое EVSE. Это напряжение между одной фазой и нейтралью. Если EVSE поддерживает многофазную зарядку, то EV должно рассчитать напряжение между фазами. Этот параметр также используется как значение для расчета соответствующего максимального зарядного тока на базе значений PMax в элементах SASchedule
EVSEMaxCurrent	complexType: PhysicalValueType см. 8.5.2.7	Максимально допустимое ограничение тока на фазу, установленное EVSE. Если коэффициент заполнения PWM-сигнала установлен в значение 5 %, то это единственное ограничение тока, принимаемое во внимание EVCC. В противном случае EVCC применяет наименьшее ограничение из значений, определяемых EVSEMaxCurrent и коэффициентом заполнения PWM-сигнала

[V2G2-708] EVSE должно рассчитывать значения PMax в SASchedule с учетом параметра EVSENominalVoltage.

[V2G2-709] EV должно использовать значение EVSENominalVoltage, направленное оборудованием EVSE, для расчета максимального тока заряда на базе значений PMax в элементе SASchedule сообщения ChargeParameterDiscoveryResponse и значений ChargingProfileEntryMaxPower в элементе ChargingProfile сообщения Power Delivery Request.

8.5.4 DC

8.5.4.1 DC_EVSEStatusType

[V2G2-364] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 90.

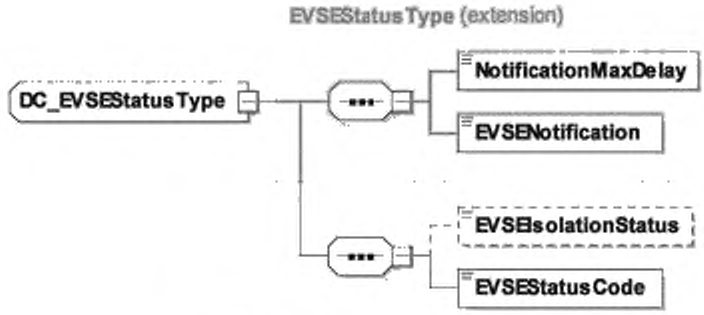


Рисунок 90 — Диаграмма — DC_EVSEStatusType

[V2G2-365] Элемент сообщения должен использоваться, как определено в таблице 96.

Таблица 96 — Семантика и определение типа для DC_EVSEStatusType

Имя элемента	Тип	Семантика
NotificationMaxDelay	simpleType: unsignedShort этот элемент унаследован от EVStatusType см. определение типа в F.6 (приложение F)	Эта величина определяет время в секундах от момента отправления данного сообщения (относительное время) до момента, когда ожидается выполнение действия. SECC использует элемент NotificationMaxDelay в EVSEStatus для указания на время, до наступления которого он ожидает реакции EVCC на запрос действия, указанного в EVSENotification
EVSENotification	simpleType: EVSENotificationType Enumeration этот элемент унаследован от EVStatusType см. определение типа в F.6 (приложение F)	Это значение используется SECC для определения реакции EVCC. EVSENotification содержит действие, выполнение которого SECC требует от EVCC. Ожидается, что запрошенное действие будет выполнено EVCC до наступления момента времени, указанного в NotificationMaxDelay. Если не подразумевается, что это отложенная временная цель, то ожидается, что EVCC выполнит действие немедленно. Во время работы значение EVSENotification установлено в значение «none» (не указано)
EVSEIsolationStatus	simpleType: isolationLevelType enumeration см. определение типа в F.6 (приложение F)	Факультативно: указывает состояние изоляции (результат мониторинга изоляции). Параметр должен обрабатываться в соответствии с *[9]*
EVSEStatusCode	simpleType: DC_EVSEStatusCodeType enumeration см. определение типа в F.6 (приложение F)	Показывает внутреннее состояние EVSE. См. подробности в таблице 98

[V2G2-801] EVCC должен использовать EVSEIsolationStatus, как описано в таблице 97.

Таблица 97 — Семантика и определение типа для isolationLevelType

Имя элемента	Семантика
Invalid	Испытание изоляции не выполнено
Valid	Испытание изоляции успешно выполнено без сигнала предупреждения или неисправности

Окончание таблицы 97

Имя элемента	Семантика
Warning	Измеренное сопротивление изоляции ниже предупредительного уровня
Fault	Измеренное сопротивление изоляции ниже уровня отказа
No_IMD	EVSE не имеет встроенного устройства мониторинга и поэтому не может выполнить испытание изоляции

[V2G2-366] SECC должен использовать коды состояния EVSEStatusCode, как описано в таблице 98, если был выбран любой из наборов сообщений DC Message Set(s), как указано в 8.6.2.

Таблица 98 — Семантика и определение типа для EVSEStatusCodeType

Имя элемента	Семантика
EVSE_NotReady	Использование не разрешено, режим ожидания, на обслуживании
EVSE_Ready	Осуществляется процедура зарядки
EVSE_Shutdown	Отключение EVSE, отключение, инициированное клиентом
EVSE_UtilityInterruptEvent	Случай сбоя электросети, оператор электросети или оборудования запросил временное снижение нагрузки
EVSE_IsolationMonitoringActive	После того как зарядная станция подтвердила внутреннюю проверку высоковольтной изоляции, она будет находиться в этом состоянии, пока проверяется целостность изоляции
EVSE_EmergencyShutdown	Несовместимость зарядных систем, аварийное отключение или нажата кнопка «Стоп» на зарядной станции
EVSE_Malfunction	Произошел неустранимый отказ зарядной станции (повреждение изоляции)
Reserved 8-C	Зарезервировано ИСО/МЭК для использования в будущем

8.5.4.2 DC_EVStatusType

[V2G2-367] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 91.

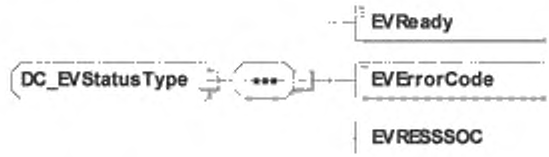


Рисунок 91 — Диаграмма — DC_EVStatusType

[V2G2-368] Элемент сообщения должен использоваться, как определено в таблице 99.

Таблица 99 — Семантика и определение типа для DC_EVStatusType

Имя элемента	Тип	Семантика
EVReady	simpleType: boolean см. определение типа в F.6 (приложение F)	Если установлено в состояние TRUE, EV готово к зарядке
EVErrorCode	simpleType: DC_EVErrorCodeType enumeration см. определение типа в F.6 (приложение F)	Показывает внутренний статус EV. См. подробности в таблице 100
EVRESSSOC	simpleType: percentValueType байт (диапазон значений: 0—100) см. определение типа в F.6 (приложение F)	Состояние зарядки, перезаряжаемой энергоаккумулирующей системы EV

[V2G2-369] EVCC должен использовать коды EVErrorCode неисправностей EV, как указано в таблице 100, если был выбран любой из наборов сообщений DC Message Set, как указано в 8.6.2.

Таблица 100 — Семантика и определение типа для EVErrorCodeType

Имя элемента	Семантика
NO_ERROR	Значение по умолчанию, когда нет обнаруженных кодов неисправностей EVCC
FAILED_RESSTemperatureInhibit	Блокировка, связанная с температурой аккумуляторной батареи. Батарея слишком горячая или слишком холодная
FAILED_EVShiftPosition	Положение селектора транспортного средства — не «Паркинг»
FAILED_ChargerConnectorLockFault	Неисправность блокировки зарядного разъема, EV не обнаружило присоединенного разъема или неисправность, при которой разъем не может быть отсоединен
FAILED_EVRESSMalfunction	Неисправность аккумуляторной батареи (RESS). Невосстанавливаемая неисправность или ошибка системы
FAILED_ChargingCurrentdifferential	Отклонение величины зарядного тока. Указание на то, что EV прервало сеанс зарядки после обнаружения неспособности зарядной станции выполнять текущий запрос
FAILED_ChargingVoltageOutOfRange	Зарядное напряжение вне границ требуемого диапазона. Индикация остановки EV сеанса зарядки после обнаружения, что напряжение аккумуляторной батареи (RESS) либо ниже, либо выше рабочего диапазона
Reserved A-C	Зарезервировано ИСО/МЭК для использования в будущем
FAILED_ChargingSystemIncompatibility	EV определяет, что зарядная станция несовместима. Использование этого показателя факультативно; как альтернативу EV может использовать EVReady в DC_EVStatusType, равный «FALSE»
NoData	Нет данных. Используется, только когда EV еще не определило свое рабочее состояние

8.5.4.3 DC_EVChargeParameterType

[V2G2-370] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 92.

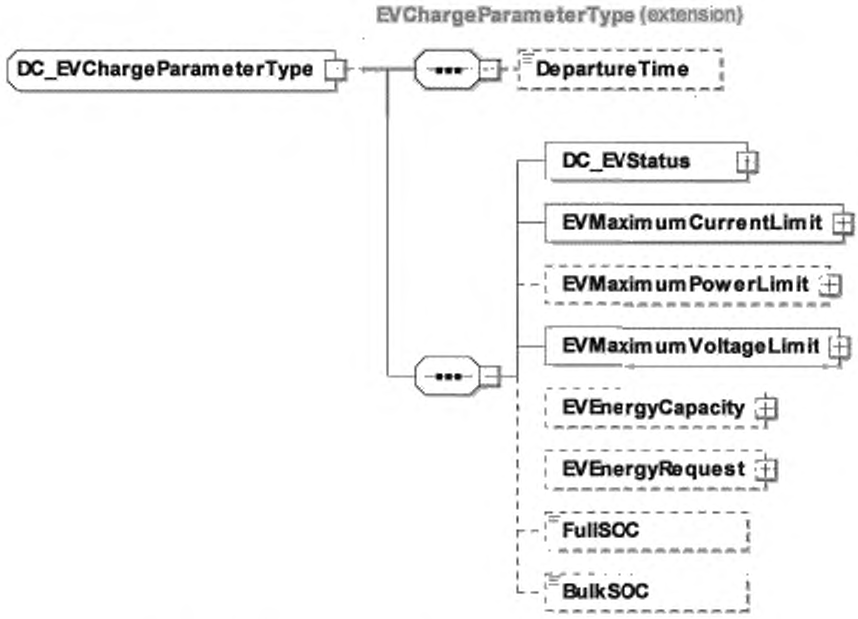


Рисунок 92 — Диаграмма — DC_EVChargeParameterType

[V2G2-371] Элемент сообщения должен использоваться, как определено в таблице 101.

Таблица 101 — Семантика и определение типа для DC_EVChargeParameterType

Имя элемента	Тип	Семантика
DepartureTime	simpleType: unsignedInt этот элемент унаследован от EVChargeParameterType см. определение типа в F.6 (приложение F)	Факультативно: этот элемент используется для указания на время, когда EV намерен завершить процесс зарядки. Представляет собой временной интервал в секундах от момента отправки данного сообщения
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока заряда EV
EVMaximumCurrentLimit	complexType: PhysicalValueType см. 8.5.2.7	Максимальный ток, поддерживаемый EV
EVMaximumPowerLimit	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: максимальная мощность, поддерживаемая EV
EVMaximumVoltageLimit	complexType: PhysicalValueType см. 8.5.2.7	Максимальное напряжение, поддерживаемое EV
EVEnergyCapacity	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: энергетическая емкость EV
EVEnergyRequest	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: количество энергии, запрашиваемое EV у EVSE

Окончание таблицы 101

Имя элемента	Тип	Семантика
FullSOC	simpleType: unitMultiplierType byte (range: 0-100) см. определение типа в F.6 (приложение F)	Факультативно: состояние зарядки, при котором EV считает аккумулятор полностью заряженным
BulkSOC	simpleType: unitMultiplierType byte (range: 0-100) см. определение типа в F.6 (приложение F)	Факультативно: состояние зарядки, при котором EV считает, что процесс быстрой зарядки должен быть завершен

8.5.4.4 DC_EVSEChargeParameterType

[V2G2-372] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 93.

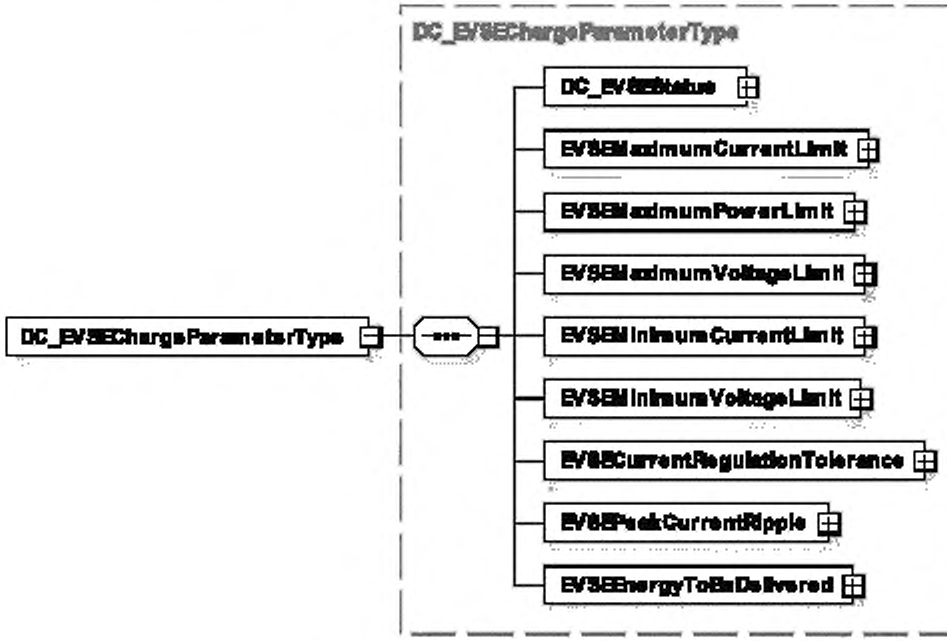


Рисунок 93 — Диаграмма — DC_EVSEChargeParameterType

[V2G2-373] Элемент сообщения должен использоваться, как определено в таблице 102.

Таблица 102 — Семантика и определение типа для DC_EVSEChargeParameterType

Имя элемента	Тип	Семантика
DC_EVSEStatus	complexType: DC_EVSEStatusType см. 8.5.4.1	Актуальное значение тока зарядки EVSE
EVSEMaximumCurrent Limit	complexType: PhysicalValueType см. 8.5.2.7	Максимальный ток, обеспечиваемый EVSE

Окончание таблицы 102

Имя элемента	Тип	Семантика
EVSEMaximumPower Limit	complexType: PhysicalValueType см. 8.5.2.7	Максимальная мощность, обеспечиваемая EVSE
EVSEMaximumVoltage Limit	complexType: PhysicalValueType см. 8.5.2.7	Максимальное напряжение, обеспечиваемое EVSE
EVSEMinimumCurrent Limit	complexType: PhysicalValueType см. 8.5.2.7	Минимальный ток, обеспечиваемый EVSE с ожидаемой точностью
EVSEMinimumVoltage Limit	complexType: PhysicalValueType см. 8.5.2.7	Минимальное напряжение, обеспечиваемое EVSE с ожидаемой точностью
EVSECurrentRegulation Tolerance	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: абсолютная величина допуска регулирования EVSE
EVSEPeakCurrentRipple	complexType: PhysicalValueType см. 8.5.2.7	Абсолютная величина размаха пульсаций тока EVSE
EVSEEnergyToBe Delivered	complexType: PhysicalValueType см. 8.5.2.7	Факультативно: количество энергии, требуемое от EVSE

8.5.4.5 DC_EVPowerDeliveryParameterType

[V2G2-374] В зависимости от выбранного(ых) набора(ов) сообщений, описанного(ых) в 8.6.2, EVCC и SECC должны реализовывать данный тип, как определено в таблице 104 и в соответствии с рисунком 94.

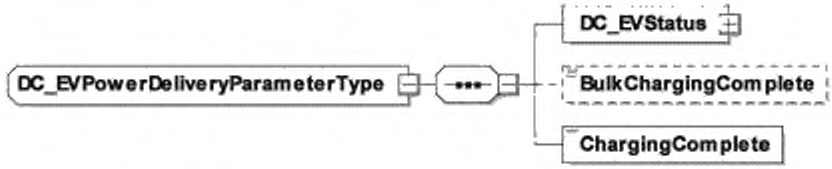


Рисунок 94 — Диаграмма — DC_EVPowerDeliveryParameterType

[V2G2-375] Элемент сообщения должен использоваться, как определено в таблице 103.

Таблица 103 — Семантика и определение типа для DC_EVPowerDeliveryParameterType

Имя элемента	Тип	Семантика
DC_EVStatus	complexType: DC_EVStatusType см. 8.5.4.2	Актуальное значение тока зарядки EV
BulkChargingComplete	simpleType: boolean см. определение типа в F.6 (приложение F)	Факультативно: если установлено в состояние TRUE, EV указывает, что основная зарядка (состояние зарядки около 80 %) выполнена
ChargingComplete	simpleType: boolean см. определение типа в F.6 (приложение F)	Если установлено в состояние TRUE, EV указывает, что полная зарядка (состояние зарядки 100 %) выполнена

8.6 Идентификационные режимы и определения наборов сообщений

8.6.1 Обзор

В *ГОСТ Р 58122 (ИСО 15118-1:2013)* описаны элементы случаев использования связи, относящиеся к разным сценариям зарядки. В зависимости от случая использования и сценария зарядки только один набор сообщений и параметров, описанный в настоящем стандарте, должен поддерживаться и передаваться EVCC и SECC.

В данном подразделе определены обязательные и опционные сообщения применительно к сценариям зарядки, использующим внешние средства идентификации (EIM), и к сценариям зарядки, использующим для идентификации режим PnC. В дальнейшем на них делается ссылка как на идентификационный режим.

На рисунке 95 показана взаимосвязь между идентификационными режимами PnC и EIM для сообщений и параметров, описанных в настоящем стандарте. В то время как идентификационный режим PnC может использовать все определения настоящего стандарта, режим EIM использует набор, из которого исключаются сообщения для поддержания режима PnC.

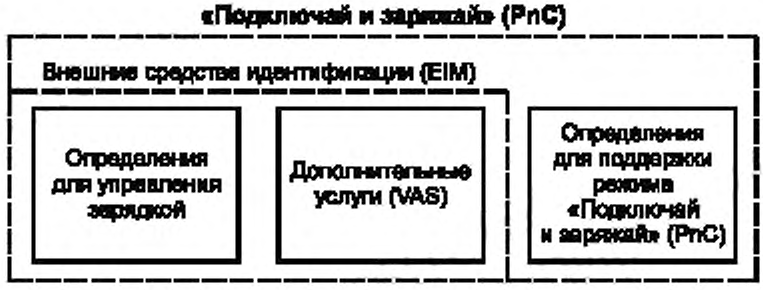


Рисунок 95 — Взаимосвязь идентификационных режимов и определений сообщений и параметров

Примечание 1 — Поскольку главным отличием между идентификационными режимами PnC и EIM является поддержка определений для режима PnC, реализация настоящего стандарта для сценария зарядки с идентификационным режимом PnC может также поддерживать такой же сценарий зарядки с идентификационным режимом EIM.

Примечание 2 — Если EV и EVSE поддерживают идентификационные режимы PnC и EIM, EV свободно в выборе предпочтительного идентификационного режима.

Ниже даются определения обязательных сообщений и параметров, охватывающих случаи использования и сценарии зарядки, которые описаны в *ГОСТ Р 58122 (ИСО15118-1:2013)*. В приложении К дано детальное описание элементов случаев использования, определенных в *ГОСТ Р 58122 (ИСО15118-1: 2013)*, и обязательных сообщений V2G и параметров, как указано в следующих пунктах.

В настоящем стандарте описано различие между обязательной/факультативной поддержкой сообщений, параметрами EVCC и SECC, а также обязательными/факультативными атрибутами параметров в определениях XML-схемы:

- обязательная/факультативная поддержка: поддержка обязательных сообщений и параметров определяет набор сообщений и параметров, которые EVCC и SECC могут обрабатывать. Это понятие обеспечивает хорошо известный набор функциональных возможностей и соответственно совместимость в отношении набора случаев использования;
- определения обязательных/факультативных параметров XML-схем: определение обязательных и факультативных параметров в XML-схеме исходит из обязательной/факультативной поддержки параметров EVCC и SECC для всех случаев использования. Если параметр обязательный во всех случаях использования, он определен как обязательный в XML-схеме. Если параметр является факультативным хотя бы в одном случае использования, то в XML-схеме он определяется как факультативный. Это позволяет отсылать сообщения только с требуемыми параметрами, опуская параметры, не являющиеся обязательными для случая использования.

Примечание 3 — Обязательная поддержка сообщения или параметра в одном блоке обмена V2G не означает, что это сообщение или параметр будет передаваться. Например, если поддержка параметра в EVCC

является факультативной и тот же самый параметр определен как обязательный в SECC, EVCC может принимать решение об отсылке данного параметра. Обязательная поддержка этого параметра в SECC говорит о том, что SECC способен обрабатывать это значение, если оно прислано EVCC. Если параметр определен как факультативный для обеих сторон, этот параметр может использоваться, только если обе стороны его поддерживают.

Примечание 4 — Основной целью определения обязательных и факультативных параметров XML-схемы является оптимизация передачи сообщений V2G. В общем случае невозможно определить обязательные и факультативные параметры, которые должны поддерживаться EVCC и SECC. Например, если поддержка двух параметров EVCC обязательна, а поддержка SECC обязательна только для одного параметра, SECC может решить послать в сообщении-ответе только один параметр. Для такого случая использования XML-схема устанавливает один факультативный параметр и один обязательный. Но XML-схема не предусматривает решения о том, на какой стороне поддержка двух параметров обязательна или факультативна для того или иного случая использования.

XML-схема со всеми обязательными и факультативными определениями подробно описана в 8.3 и в приложении F. Следующие подпункты определяют обязательные и факультативные сообщения и параметры, которые должны поддерживаться EVCC и SECC, а также наборы сообщений SECC для идентификационных режимов PnC и EIM.

Набор сообщений определяет обязательные сообщения и параметры для EVCC и SECC, охватывая один или несколько элементов случаев использования, описанных в *ГОСТ Р 58122 (ИСО 15118-1:2013)*. Идентификационный режим объединяет один или несколько обязательных и факультативных наборов сообщений, охватывающих набор схожих сценариев зарядки. Идентификационный режим состоит не менее чем из одного обязательного набора сообщений, обеспечивающего основу для конкретного набора сценариев зарядки. Идентификационный режим может дополнительно использовать один или несколько факультативных наборов сообщений, обеспечивающих расширение сценариев зарядки.

На рисунке 96 дан обзор идентификационных режимов и наборов сообщений. Идентификационный режим EIM определяет сообщения и параметры для сценариев зарядки с авторизацией вне EV, как описано в *ГОСТ Р 58122 (ИСО 15118-1:2013)*. Идентификационный режим PnC определяет сообщения и параметры для сценариев зарядки с авторизацией в EV, как описано в *ГОСТ Р 58122 (ИСО 15118-1:2013)*.

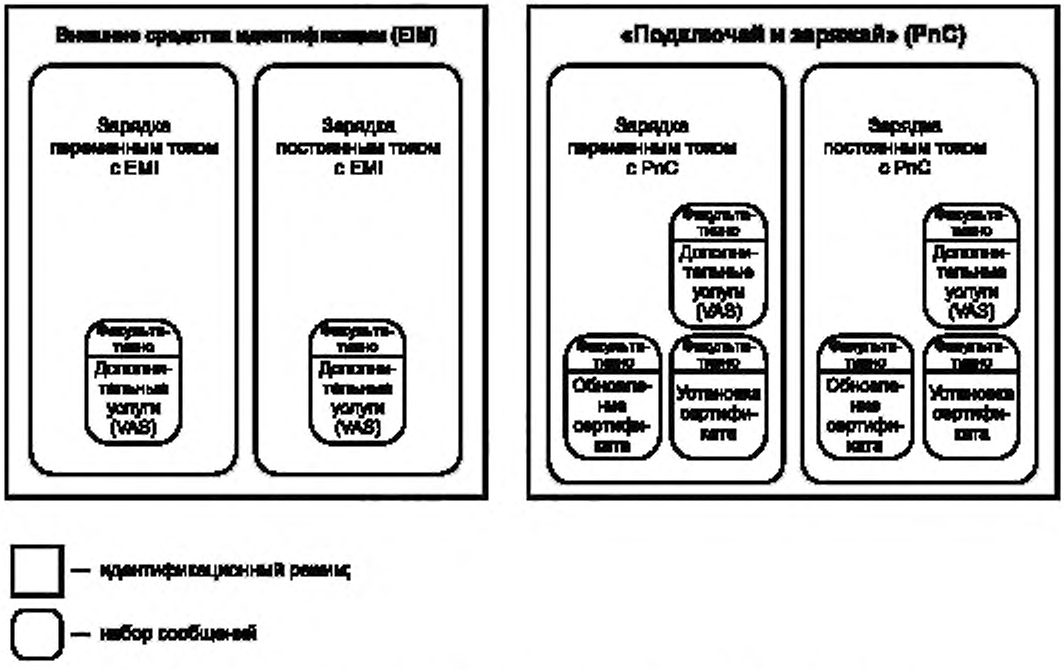


Рисунок 96 — Обзор идентификационных режимов и наборов сообщений

Настоящий стандарт определяет четыре набора сообщений, применяемых в зависимости от идентификационного режима (EIM, PnC) и режима зарядки (зарядка переменным или постоянным током):

- зарядка переменным током с EIM;
- зарядка постоянным током с EIM;
- зарядка переменным током с PnC;
- зарядка постоянным током с PnC.

Дополнительно четыре набора сообщений могут быть расширены факультативными наборами сообщений, определяющими необходимые обязательные и факультативные параметры, а также сообщениями для поддержания дополнительных случаев использования. Факультативные наборы сообщений могут отменять определения базовых наборов сообщений.

[V2G2-760] Если факультативный набор сообщений определяет параметры, которые уже определены в наборе сообщений, применяется определение факультативного набора сообщений.

[V2G2-828] Если EV, которое соответствует *[1], ГОСТ 58122 (ИСО 15118-1:2013)*, настоящему стандарту и поддерживает набор сообщений «Зарядка переменным током с EIM», определяет номинальное значение коэффициента заполнения до сообщения PaymentServiceSelectionReq, то оно должно указывать «ExternalPayment» в параметре SelectedPaymentOption в сообщении PaymentServiceSelectionReq, как определено в 8.4.3.5.

8.6.2 Поддерживаемые наборы сообщений

8.6.2.1 Обзор поддерживаемых наборов сообщений

Изготовитель EV и изготовитель EVSE могут выбирать между поддержкой одного или нескольких наборов сообщений, как показано на рисунке 96. В таблице 104 даны определения всех обязательных частей наборов сообщений, показанных на рисунке 96.

Для EVCC действуют следующие требования.

[V2G2-659] EVCC должен поддерживать сообщение или параметр в наборе сообщений, если он отмечен для EVCC как «M» (обязательный).

[V2G2-660] Если требованиями не определено другое, EVCC может поддерживать параметр в наборе сообщений, если он отмечен для EVCC как «O» (факультативный).

[V2G2-661] EVCC не должен поддерживать какой-либо параметр в наборе сообщений, если он отмечен для EVCC знаком «–».

[V2G2-662] EVCC должен передавать параметр, если он отмечен для EVCC знаком «M» в сообщении-запросе.

[V2G2-663] EVCC должен поддерживать обработку параметра, если он отмечен для EVCC знаком «M» в сообщении-ответе.

В общем случае к SECC применимы следующие требования:

[V2G2-664] SECC должен поддерживать сообщение или параметр в наборе сообщений, если он отмечен для SECC знаком «M».

[V2G2-665] Если требованиями не определено другое, SECC может поддерживать параметр в наборе сообщений, если он отмечен для SECC знаком «O».

[V2G2-666] SECC не должен поддерживать какой-либо параметр в наборе сообщений, если он отмечен для SECC знаком «–».

[V2G2-667] SECC должен передавать параметр, если он отмечен для SECC знаком «M» в сообщении-ответе.

[V2G2-668] SECC должен поддерживать обработку параметра, если он отмечен знаком «M» для SECC в сообщении-ответе.

Таблица 104 — Обязательные сообщения и элементы сообщений наборов сообщений

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опрос: Обязательной сертификат		Опция: Установочный сертификат		Общее: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/ BC	EVCC	SECC / SA	EVCC	SECC
supportedAppProtocolReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ProtocolNamespace	M	M	M	M	M	M	M	M	–	–	–	–	–	–
VersionNumberMajor	M	M	M	M	M	M	M	M	–	–	–	–	–	–
VersionNumberMinor	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SchemaID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Priority	M	M	M	M	M	M	M	M	–	–	–	–	–	–
supportedAppProtocolRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SchemaID	M	O	M	O	M	O	M	O	–	–	–	–	–	–
SessionSetupReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
EVCCID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionSetupRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECD / SA	EVCC	SECC
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
EVSEID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
DateTimeNow	O	O	O	O	O	O	O	O	–	–	–	–	–	–
ServiceDiscoveryReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceScope	O	O	O	O	O	O	O	O	–	–	–	–	–	–
ServiceCategory	O	M	O	M	O	M	O	M	–	–	–	–	–	–
ServiceDiscoveryRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PaymentOptions	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PaymentOption	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PaymentOption	M	O	M	O	M	O	M	O	–	–	–	–	–	–
ChargeService	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceName	O	O	O	O	O	O	O	O	–	–	–	–	–	–
ServiceCategory	O	M	O	M	O	M	O	M	–	–	–	–	–	–
ServiceScope	O	O	O	O	O	O	O	O	–	–	–	–	–	–
FreeService	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SupportedEnergyTransferMode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
EnergyTransferMode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceList	O	O	O	O	O	O	O	O	M	M	M	M	M	M

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Service	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceName	O	O	O	O	O	O	O	O	–	–	–	–	–	–
ServiceCategory	O	M	O	M	O	M	O	M	–	–	–	–	–	–
ServiceScope	O	O	O	O	O	O	O	O	–	–	–	–	–	–
FreeService	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceDetailReq	O	O	O	O	O	O	O	O	M	M	M	M	M	M
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceDetailRes	O	O	O	O	O	O	O	O	M	M	M	M	M	M
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServiceParameterList	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ParameterSet	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ParameterSetID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Parameter	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServicePaymentSelectionReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SelectedPaymentOption	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SelectedServiceList	O	O	O	O	O	O	O	O	M	M	M	M	M	M
SelectedService	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
ServiceID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ParameterSetID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ServicePaymentSelectionRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PaymentDetailsReq	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Header	–	–	–	–	M	M	M	M	–	–	–	–	–	–
SessionId	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Notification	–	–	–	–	O	M	O	M	–	–	–	–	–	–
FaultCode	–	–	–	–	M	M	M	M	–	–	–	–	–	–
FaultMsg	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	–	–	–	–	M	M	M	M	–	–	–	–	–	–
eMAID	–	–	–	–	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
ContractSignatureCertChain	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Certificate	–	–	–	–	M	M	M	M	–	–	–	–	–	–
SubCertificates	–	–	–	–	O	O	O	O	–	–	–	–	–	–
Certificate	–	–	–	–	M	M	M	M	–	–	–	–	–	–
PaymentDetailsReq	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Header	–	–	–	–	M	M	M	M	–	–	–	–	–	–
SessionId	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Notification	–	–	–	–	O	M	O	M	–	–	–	–	–	–
FaultCode	–	–	–	–	M	M	M	M	–	–	–	–	–	–
FaultMsg	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	–	–	–	–	M	M	M	M	–	–	–	–	–	–
ResponseCode	–	–	–	–	M	M	M	M	–	–	–	–	–	–
GenChallenge	–	–	–	–	M	M	M	M	–	–	–	–	–	–
DateTimeNow	–	–	–	–	M	M	M	M	–	–	–	–	–	–
AuthorizationReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM			Зарядка постоянным током с EIM			Зарядка переменным током с PnC			Зарядка постоянным током с PnC			Опция: Обязательная сертификация			Опция: Установленная сертификация			Опция: Установленная сертификация			Общее УШ		
	E	S	V	E	S	V	E	S	V	E	S	V	E	S	V	E	S	V	E	S	V	E	S	V
	O	O	O	O	O	O	O	O	O	O	O	O	O	O	/ SAP ID	O	O	/ SA	O	O	/ SA			
FaultMsg	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Signature (ом. отдельную таблицу)	–	–	–	–	–	–	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
GenChallenge	–	–	–	–	–	–	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
AuthorizationReq	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Notification	O	M	M	O	M	M	O	M	M	O	M	M	–	–	–	–	–	–	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Signature (ом. отдельную таблицу)	–	–	–	–	–	–	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
EVSEProcessing	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
ChargeParameterDiscoveryReq	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
Notification	O	M	M	O	M	M	O	M	M	O	M	M	–	–	–	–	–	–	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	M	M	M	M	–	–	–	–	–	–	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
MeteringSASchedulesTuple	O	M	O	M	O	M	O	M	–	–	–	–	–	–
RequestedEnergyTransferMode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
AC_EVChargeParameter	M	M	–	–	M	M	–	–	–	–	–	–	–	–
DepartureTime	O	M	–	–	O	M	–	–	–	–	–	–	–	–
Eamount	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVMaxVoltage	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVMaxCurrent	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVMinCurrent	M	M	–	–	M	M	–	–	–	–	–	–	–	–
AC_EVChargeParameter	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DepartureTime	–	–	O	M	–	–	O	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVMaximumCurrentLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVMaximumPowerLimit	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVMaximumVoltageLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
EVEnergyCapacity	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVEnergyRequest	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FullSOC	–	–	O	M	–	–	O	M	–	–	–	–	–	–
BulkSOC	–	–	O	M	–	–	O	M	–	–	–	–	–	–
ChargeParameterDiscoveryRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
EVSEProcessing	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SASchedule	M	O	M	O	M	O	M	O	–	–	–	–	–	–
SAScheduleTuple	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SAScheduleTupleID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PMaxSchedule	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PMaxScheduleID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
PMaxScheduleEntry	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
RelativeTimeInterval	M	M	M	M	M	M	M	M	–	–	–	–	–	–
start	M	M	M	M	M	M	M	M	–	–	–	–	–	–
duration	M	O	M	O	M	O	M	O	–	–	–	–	–	–
Pmax	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SalesTariff	M	O	M	O	M	O	M	O	–	–	–	–	–	–
SalesTariffID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SalesTariffDescription	O	O	O	O	O	O	O	O	–	–	–	–	–	–
NumEPriceLevels	M	O	M	O	M	O	M	O	–	–	–	–	–	–
SalesTariffEntry	M	M	M	M	M	M	M	M	–	–	–	–	–	–
RelativeTimeInterval	M	M	M	M	M	M	M	M	–	–	–	–	–	–
start	M	M	M	M	M	M	M	M	–	–	–	–	–	–
duration	M	O	M	O	M	O	M	O	–	–	–	–	–	–
EPriceLevel	M	O	M	O	M	O	M	O	–	–	–	–	–	–
ConsumptionCost	M	O	M	O	M	O	M	O	–	–	–	–	–	–
Start Value	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Cost	M	O	M	O	M	O	M	O	–	–	–	–	–	–
AC_EVSEChargeParameter	M	M	–	–	M	M	–	–	–	–	–	–	–	–
AC_EVSEStatus	M	M	–	–	M	M	–	–	–	–	–	–	–	–
RCD	M	M	–	–	M	M	–	–	–	–	–	–	–	–
NotificationMaxDelay	M	M	–	–	M	M	–	–	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опции: Обновление сертификата		Опции: Установка сертификата		Опции: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/ BC	EVCC	SECC / SA	EVCC	SECC
EVSENotification	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSENominalVoltage	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSEMaxCurrent	M	M	–	–	M	M	–	–	–	–	–	–	–	–
DC_EVSEChargeParameter	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSE Status Code	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMaximumCurrentLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMaximumPowerLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMaximumVoltageLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMinimumCurrentLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMinimumVoltageLimit	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSECurrentRegulationTolerance	–	–	O	O	–	–	O	O	–	–	–	–	–	–
EVSEPeakCurrentRipple	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEEnergyToBeDelivered	–	–	O	O	–	–	O	O	–	–	–	–	–	–
PowerDeliveryReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargeProgress	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SASchedule TupleID	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingProfile	O	M	O	M	O	M	O	M	–	–	–	–	–	–
ProfileEntry	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingProfileEntryStart	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingProfileEntryMaxPower	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingProfileEntryMaxNumberOfPhasesInUse	O	M	–	–	O	M	–	–	–	–	–	–	–	–
DC_EVPowerDeliveryParameter	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
BulkChargingComplete	–	–	O	M	–	–	O	M	–	–	–	–	–	–
ChargingComplete	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECD / SA	EVCC	SECC
PowerDeliveryReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
AC_EVSEStatus	M	M	–	–	M	M	–	–	–	–	–	–	–	–
RCD	M	M	–	–	M	M	–	–	–	–	–	–	–	–
NotificationMaxDelay	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSENotification	M	M	–	–	M	M	–	–	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEStatus Code	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
CertificateUpdateReq	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Header	–	–	–	–	–	–	–	–	M	M	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Откл: Обновление сертификата		Откл: Установка сертификата		Общее: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
SessionId	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Notification	–	–	–	–	–	–	–	–	M	M	–	–	–	–
FaultCode	–	–	–	–	–	–	–	–	M	M	–	–	–	–
FaultMsg	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Body	–	–	–	–	–	–	–	–	M	M	–	–	–	–
ContractSignatureCertChain	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Certificate	–	–	–	–	–	–	–	–	M	M	–	–	–	–
SubCertificates	–	–	–	–	–	–	–	–	O	O	–	–	–	–
Certificate	–	–	–	–	–	–	–	–	M	M	–	–	–	–
eMAID	–	–	–	–	–	–	–	–	M	M	–	–	–	–
ListOfRootCertificateIDs	–	–	–	–	–	–	–	–	M	M	–	–	–	–
RootCertificateID	–	–	–	–	–	–	–	–	M	M	–	–	–	–
CertificateUpdateRes	–	–	–	–	–	–	–	–	O	O	–	–	–	–
Header	–	–	–	–	–	–	–	–	M	M	–	–	–	–
SessionId	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Notification	–	–	–	–	–	–	–	–	O	M	–	–	–	–
FaultCode	–	–	–	–	–	–	–	–	M	M	–	–	–	–
FaultMsg	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	O	M	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Body	–	–	–	–	–	–	–	–	M	M	–	–	–	–
ResponseCode	–	–	–	–	–	–	–	–	M	M	–	–	–	–
SAProvisioningCertificateChain	–	–	–	–	–	–	–	–	M	M	–	–	–	–
ContractSignatureCertChain	–	–	–	–	–	–	–	–	M	M	–	–	–	–
Certificate	–	–	–	–	–	–	–	–	M	M	–	–	–	–
SubCertificates	–	–	–	–	–	–	–	–	O	O	–	–	–	–
Certificate	–	–	–	–	–	–	–	–	M	M	–	–	–	–
ContractSignatureEncryptedPrivateKey	–	–	–	–	–	–	–	–	M	M	–	–	–	–
DHpublickey	–	–	–	–	–	–	–	–	M	M	–	–	–	–
eMAID	–	–	–	–	–	–	–	–	M	M	–	–	–	–
RetryCounter	–	–	–	–	–	–	–	–	M	O	–	–	–	–
CertificateInstallationReq	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Header	–	–	–	–	–	–	–	–	–	–	M	M	–	–
SessionId	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Notification	–	–	–	–	–	–	–	–	–	–	O	M	–	–
FaultCode	–	–	–	–	–	–	–	–	–	–	M	M	–	–
FaultMsg	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Body	–	–	–	–	–	–	–	–	–	–	M	M	–	–
OEMProvisioningCert	–	–	–	–	–	–	–	–	–	–	M	M	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
ListOfRootCertificateIDs	–	–	–	–	–	–	–	–	–	–	M	M	–	–
RootCertificateID	–	–	–	–	–	–	–	–	–	–	M	M	–	–
CertificateInstallationRes	–	–	–	–	–	–	–	–	–	–	O	O	–	–
Header	–	–	–	–	–	–	–	–	–	–	M	M	–	–
SessionId	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Notification	–	–	–	–	–	–	–	–	–	–	M	M	–	–
FaultCode	–	–	–	–	–	–	–	–	–	–	M	M	–	–
FaultMsg	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	O	M	–	–
Body	–	–	–	–	–	–	–	–	–	–	M	M	–	–
ResponseCode	–	–	–	–	–	–	–	–	–	–	M	M	–	–
SAProvisioningCertificateChain	–	–	–	–	–	–	–	–	–	–	M	M	–	–
ContractSignatureCertChain	–	–	–	–	–	–	–	–	–	–	M	M	–	–
Certificate	–	–	–	–	–	–	–	–	–	–	M	M	–	–
SubCertificates	–	–	–	–	–	–	–	–	–	–	O	O	–	–
Certificate	–	–	–	–	–	–	–	–	–	–	M	M	–	–
ContractSignatureEncryptedPrivateKey	–	–	–	–	–	–	–	–	–	–	M	M	–	–
DhPublickey	–	–	–	–	–	–	–	–	–	–	M	M	–	–
eMAID	–	–	–	–	–	–	–	–	–	–	M	M	–	–
SessionStopReq	M	M	M	M	M	M	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECD / SA	EVCC	SECC
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingSession	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionStopRes	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Header	M	M	M	M	M	M	M	M	–	–	–	–	–	–
SessionId	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Notification	O	M	O	M	O	M	O	M	–	–	–	–	–	–
FaultCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
FaultMsg	M	M	M	M	M	M	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	M	M	–	–	–	–	–	–
Body	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ResponseCode	M	M	M	M	M	M	M	M	–	–	–	–	–	–
ChargingStatusReq	M	M	–	–	M	M	–	–	–	–	–	–	–	–
Header	M	M	–	–	M	M	–	–	–	–	–	–	–	–
SessionId	M	M	–	–	M	M	–	–	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
Notification	O	M	–	–	O	M	–	–	–	–	–	–	–	–
FaultCode	M	M	–	–	M	M	–	–	–	–	–	–	–	–
FaultMsg	M	M	–	–	M	M	–	–	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	M	M	–	–	M	M	–	–	–	–	–	–	–	–
ChargingStatusRes	M	M	–	–	M	M	–	–	–	–	–	–	–	–
Header	M	M	–	–	M	M	–	–	–	–	–	–	–	–
SessionId	M	M	–	–	M	M	–	–	–	–	–	–	–	–
Notification	O	M	–	–	O	M	–	–	–	–	–	–	–	–
FaultCode	M	M	–	–	M	M	–	–	–	–	–	–	–	–
FaultMsg	M	M	–	–	M	M	–	–	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	M	M	–	–	M	M	–	–	–	–	–	–	–	–
ResponseCode	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSEID	M	M	–	–	M	M	–	–	–	–	–	–	–	–
SAScheduleTupleID	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSEMaxCurrent	M	O	–	–	O	O	–	–	–	–	–	–	–	–
MeterInfo	M	O	–	–	M	O	–	–	–	–	–	–	–	–
MeterID	M	M	–	–	M	M	–	–	–	–	–	–	–	–
MeterReading	O	O	–	–	O	O	–	–	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
SigMeterReading	O	O	–	–	O	O	–	–	–	–	–	–	–	–
MeterStatus	O	O	–	–	O	O	–	–	–	–	–	–	–	–
TMeter	O	O	–	–	O	O	–	–	–	–	–	–	–	–
ReceiptRequired	–	–	–	–	M	O	–	–	–	–	–	–	–	–
AC_EVSEStatus	M	M	–	–	M	M	–	–	–	–	–	–	–	–
RCD	M	M	–	–	M	M	–	–	–	–	–	–	–	–
NotificationMaxDelay	M	M	–	–	M	M	–	–	–	–	–	–	–	–
EVSENotification	M	M	–	–	M	M	–	–	–	–	–	–	–	–
MeteringReceiptReq	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Header	–	–	–	–	M	M	–	–	–	–	–	–	–	–
SessionId	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Notification	–	–	–	–	O	M	–	–	–	–	–	–	–	–
FaultCode	–	–	–	–	M	M	–	–	–	–	–	–	–	–
FaultMsg	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Body	–	–	–	–	M	M	–	–	–	–	–	–	–	–
SessionID	–	–	–	–	M	M	–	–	–	–	–	–	–	–
BASScheduleTupleID	–	–	–	–	O	M	–	–	–	–	–	–	–	–
MeterInfo	–	–	–	–	M	M	–	–	–	–	–	–	–	–
MeterID	–	–	–	–	M	M	–	–	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
MeterReading	–	–	–	–	O	O	–	–	–	–	–	–	–	–
SigMeterReading	–	–	–	–	O	O	–	–	–	–	–	–	–	–
MeterStatus	–	–	–	–	O	O	–	–	–	–	–	–	–	–
Tmeter	–	–	–	–	O	O	–	–	–	–	–	–	–	–
MeteringReceiptRes	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Header	–	–	–	–	M	M	–	–	–	–	–	–	–	–
SessionId	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Notification	–	–	–	–	O	M	–	–	–	–	–	–	–	–
FaultCode	–	–	–	–	M	M	–	–	–	–	–	–	–	–
FaultMsg	–	–	–	–	M	M	–	–	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	–	–	M	M	–	–	–	–	–	–	–	–
AC_EVSEStatus	–	–	–	–	M	M	–	–	–	–	–	–	–	–
RCD	–	–	–	–	M	M	–	–	–	–	–	–	–	–
NotificationMaxDelay	–	–	–	–	M	M	–	–	–	–	–	–	–	–
EVSENotification	–	–	–	–	M	M	–	–	–	–	–	–	–	–
DC_EVSEStatus	–	–	–	–	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	–	–	–	–	M	O	–	–	–	–	–	–
EVSEStatusCode	–	–	–	–	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	–	–	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
EVSENotification	–	–	–	–	–	–	M	M	–	–	–	–	–	–
ResponseCode	–	–	–	–	M	M	–	–	–	–	–	–	–	–
CableCheckReq	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
CableCheckRes	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
ResponseCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEProcessing	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEStatusCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
PreChargeReq	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVTargetVoltage	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVTargetCurrent	–	–	M	M	–	–	M	M	–	–	–	–	–	–
PreChargeRes	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
ResponseCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEStatusCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEPresentVoltage	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECC / SA	EVCC	SECC
CurrentDemandReq	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVTargetCurrent	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVMaximumVoltageLimit	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVMaximumCurrentLimit	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVMaximumPowerLimit	–	–	O	M	–	–	O	M	–	–	–	–	–	–
BulkChargingComplete	–	–	O	M	–	–	O	M	–	–	–	–	–	–
ChargingComplete	–	–	M	M	–	–	M	M	–	–	–	–	–	–
RemainingTimeToFullSoC	–	–	O	M	–	–	O	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECD / SA	EVCC	SECC
RemainingTimeToBulkSoC	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVTargetVoltage	–	–	M	M	–	–	M	M	–	–	–	–	–	–
CurrentDemandRes	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
ResponseCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEStatusCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEPresentVoltage	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEPresentCurrent	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSECurrentLimitAchieved	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEVoltageLimitAchieved	–	–	M	M	–	–	M	M	–	–	–	–	–	–

Продолжение таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BO	EVCC	SECC / SA	EVCC	SECC
EVSEPowerLimitAchieved	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEMaximumVoltageLimit	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEMaximumCurrentLimit	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEMaximumPowerLimit	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEID	–	–	M	M	–	–	M	O	–	–	–	–	–	–
SAScheduleTupleID	–	–	M	M	–	–	M	O	–	–	–	–	–	–
MeterInfo	–	–	–	–	–	–	M	O	–	–	–	–	–	–
MeterID	–	–	–	–	–	–	M	M	–	–	–	–	–	–
MeterReading	–	–	–	–	–	–	O	O	–	–	–	–	–	–
SigMeterReading	–	–	–	–	–	–	O	O	–	–	–	–	–	–
MeterStatus	–	–	–	–	–	–	O	O	–	–	–	–	–	–
Tmeter	–	–	–	–	–	–	O	O	–	–	–	–	–	–
ReceiptRequired	–	–	–	–	–	–	M	O	–	–	–	–	–	–
WeldingDetectionReq	–	–	O	M	–	–	O	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Окончание таблицы 104

	Зарядка переменным током с EIM		Зарядка постоянным током с EIM		Зарядка переменным током с PnC		Зарядка постоянным током с PnC		Опция: Обновление сертификата		Опция: Установка сертификата		Опция: VAS	
	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC	EVCC	SECC / SA/BC	EVCC	SECD / SA	EVCC	SECC
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVReady	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSConditioning	–	–	O	M	–	–	O	M	–	–	–	–	–	–
EVErrorCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVRESSSOC	–	–	M	M	–	–	M	M	–	–	–	–	–	–
WeldingDetectionReq	–	–	O	M	–	–	O	M	–	–	–	–	–	–
Header	–	–	M	M	–	–	M	M	–	–	–	–	–	–
SessionId	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Notification	–	–	O	M	–	–	O	M	–	–	–	–	–	–
FaultCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
FaultMsg	–	–	M	M	–	–	M	M	–	–	–	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Body	–	–	M	M	–	–	M	M	–	–	–	–	–	–
ResponseCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
DC_EVSEStatus	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEIsolationStatus	–	–	M	O	–	–	M	O	–	–	–	–	–	–
EVSEStatusCode	–	–	M	M	–	–	M	M	–	–	–	–	–	–
NotificationMaxDelay	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSENotification	–	–	M	M	–	–	M	M	–	–	–	–	–	–
EVSEPresentVoltage	–	–	M	M	–	–	M	M	–	–	–	–	–	–

8.6.2.2 Общие требования

[V2G2-762] Если SECC работает в идентификационном режиме EIM и решает направить тарифную ставку, то это должна быть тарифная ставка, выбранная EVSE (например, при использовании карты радиочастотной идентификации) в первой записи (SAScheduleTuple) в составе параметра SAScheduleList в сообщении ChargeParameterDiscoveryRes.

[V2G2-763] Если EVCC работает в идентификационном режиме EIM, он должен обработать только первую запись (SAScheduleTuple) в параметре SAScheduleList, включенном в сообщение ChargeParameterDiscoveryRes.

Примечание — В режиме PnC параметр тарифной ставки может содержать несколько тарифов, позволяя EVCC выбирать один из них для внутренней оптимизации графика зарядки. В идентификационном режиме EIM выбор тарифной ставки осуществляется оборудованием EVSE. Это означает, что SECC передает тарифную ставку EVCC и EV может использовать эту информацию для оптимизации графика зарядки.

8.6.2.3 Переменный ток

8.6.2.3.1 EIM

Если EVCC/SECC поддерживает идентификационный режим EIM для зарядки переменным током, он поддерживает соответствующие наборы сообщений.

[V2G2-376] Если EVCC поддерживает зарядку переменным током и идентификационный режим EIM, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка переменным током с EIM», графа «EVCC».

[V2G2-377] Если SECC поддерживает зарядку переменным током и идентификационный режим EIM, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка переменным током с EIM», графа «SECC».

Если EVCC/SECC поддерживает набор сообщений «Зарядка переменным током с EIM», он может дополнительно поддерживать набор сообщений «Value Added Services».

[V2G2-378] Если EVCC поддерживает набор сообщений «Зарядка переменным током с EIM», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «EVCC».

[V2G2-379] Если SECC поддерживает набор сообщений «Зарядка переменным током с EIM», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «SECC».

8.6.2.3.2 PnC

Если EVCC/SECC поддерживает идентификационный режим PnC для зарядки переменным током, он поддерживает соответствующие наборы сообщений.

[V2G2-380] Если EVCC поддерживает зарядку переменным током и идентификационный режим PnC, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка переменным током с PnC», графа «EVCC».

[V2G2-381] Если SECC поддерживает зарядку переменным током и идентификационный режим PnC, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка переменным током с PnC», графа «SECC».

Если EVCC/SECC поддерживает идентификационный режим PnC для зарядки переменным током, он может дополнительно поддерживать набор сообщений: «VAS», «Обновление сертификата» и «Установка сертификата».

[V2G2-384] Если EVCC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «EVCC».

[V2G2-385] Если SECC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «SECC».

[V2G2-386] Если EVCC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Установка сертификата», графа «EVCC».

[V2G2-387] Если SECC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Установка сертификата», графа «SECC».

[V2G2-388] Если EVCC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Обновление сертификата», графа «EVCC».

[V2G2-389] Если SECC поддерживает набор сообщений «Зарядка переменным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Обновление сертификата», графа «SECC».

8.6.2.4 Постоянный ток

8.6.2.4.1 Зарядка постоянным током с EIM

Если EVCC/SECC поддерживает идентификационный режим с EIM для зарядки постоянным током, он поддерживает соответствующие наборы сообщений.

[V2G2-390] Если EVCC поддерживает зарядку постоянным током и идентификационный режим EIM, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка постоянным током с EIM», графа «EVCC».

[V2G2-391] Если SECC поддерживает зарядку постоянным током и идентификационный режим EIM, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка постоянным током с EIM», графа «SECC».

Если EVCC/SECC поддерживает набор сообщений «Зарядка постоянным током с EIM», он может дополнительно поддерживать набор сообщений «VAS».

[V2G2-392] Если EVCC поддерживает набор сообщений «Зарядка постоянным током с EIM», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «EVCC».

[V2G2-393] Если SECC поддерживает набор сообщений «Зарядка постоянным током с EIM», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «SECC».

8.6.2.4.2 PnC

Если EVCC/SECC поддерживает идентификационный режим PnC для зарядки постоянным током, он поддерживает соответствующие наборы сообщений.

[V2G2-394] Если EVCC поддерживает зарядку постоянным током и идентификационный режим PnC, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка постоянным током с PnC», графа «EVCC».

[V2G2-395] Если SECC поддерживает зарядку постоянным током и идентификационный режим PnC, он должен поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Зарядка постоянным током с PnC», графа «SECC».

Если EVCC/SECC поддерживает идентификационный режим PnC для зарядки постоянным током, он может дополнительно поддерживать набор сообщений «Опция: VAS», «Опция: Обновление сертификата» и «Опция: Установка сертификата».

[V2G2-396] Если EVCC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «EVCC».

[V2G2-397] Если SECC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: VAS», графа «SECC».

[V2G2-398] Если EVCC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Установка сертификата», графа «EVCC».

[V2G2-399] Если SECC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Установка сертификата», графа «SECC».

[V2G2-400] Если EVCC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Обновление сертификата», графа «EVCC».

[V2G2-401] Если SECC поддерживает набор сообщений режима «Зарядка постоянным током с PnC», он может дополнительно поддерживать сообщения и параметры, отмеченные знаками «M» и «O» в таблице 104, графа «Опция: Обновление сертификата», графа «SECC».

8.6.3 Выбор наборов сообщений

8.6.3.1 Наборы сообщений для зарядки переменным/постоянным током в режимах EIM/PnC

Выбор набора сообщений основан на варианте оплаты в сообщении PaymentServiceSelectionReq.

Рисунок 97 дает общее представление выбора наборов сообщений «Зарядка переменным током с EIM», «Зарядка постоянным током с EIM», «Зарядка переменным током с PnC» и «Зарядка постоянным током с PnC» на базе определений для PaymentServiceSelectionReq (см. 8.4.3.5.2) и ChargeParameterDiscoveryReq (см. 8.4.3.8.2).

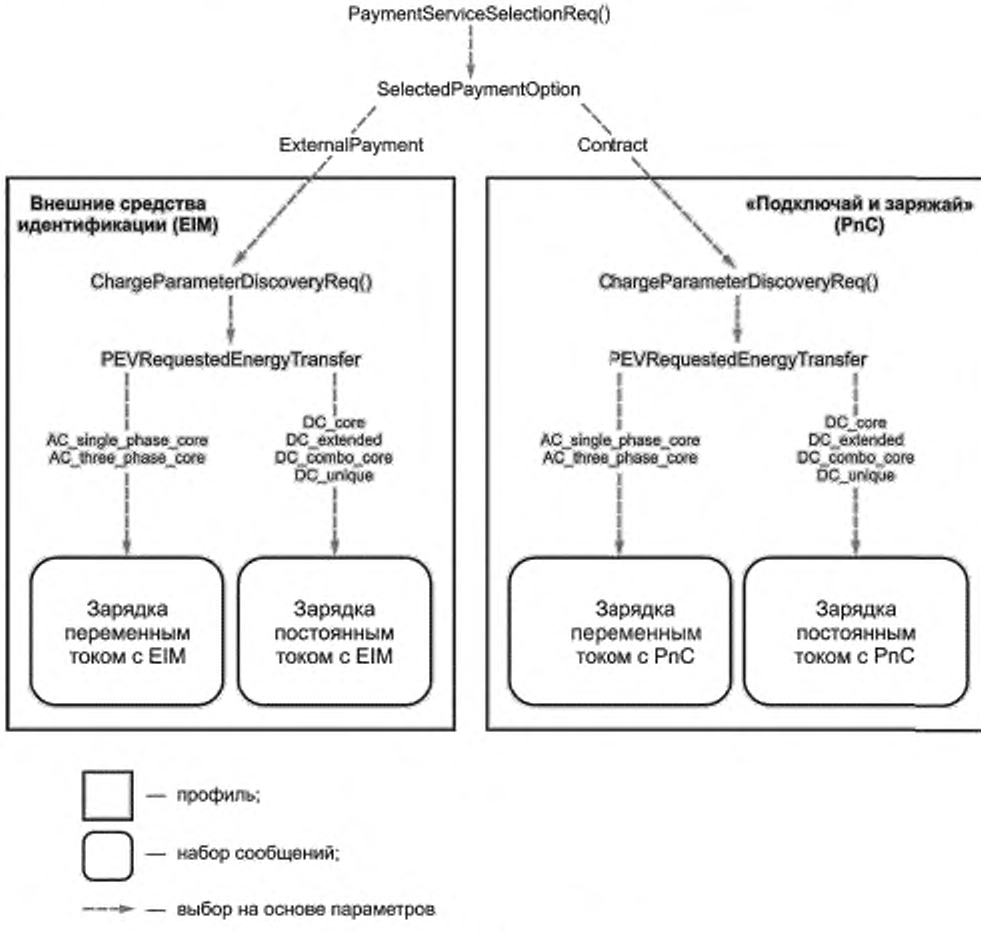


Рисунок 97 — Выбор наборов сообщений

[V2G2-402] Если EVCC отправляет опцию оплаты External Payment в параметре SelectedPaymentOption сообщения PaymentServiceSelectionReq и посылает AC_single_phase_core или AC_three_phase_core в параметре RequestedEnergyTransferMode сообщения ChargeParameterDiscoveryReq, то он должен использовать набор сообщений «Зарядка переменным током с EIM».

[V2G2-403] Если EVCC отправляет опцию оплаты «External Payment» в параметре SelectedPaymentOption сообщения PaymentServiceSelectionReq и посылает DC_core, или DC_extended, или DC_combo_core, или DC_unique в параметре RequestedEnergyTransferMode сообщения ChargeParameterDiscoveryReq, то он должен использовать набор сообщений «Зарядка постоянным током с EIM».

[V2G2-404] Если EVCC отправляет опцию оплаты «Contract» в параметре SelectedPaymentOption сообщения PaymentServiceSelectionReq и посылает AC_single_phase_core или AC_three_phase_core в параметре RequestedEnergyTransferMode сообщения ChargeParameterDiscoveryReq, то он должен использовать набор сообщений «Зарядка переменным током с PnC».

[V2G2-405] Если EVCC отправляет опцию оплаты «Contract» в параметре SelectedPaymentOption сообщения PaymentServiceSelectionReq и посылает DC_core, или DC_extended, или DC_combo_core, или DC_unique в параметре RequestedEnergyTransferMode сообщения ChargeParameterDiscoveryReq, то он должен использовать набор сообщений «Зарядка постоянным током с PnC».

8.6.3.2 Набор сообщений квитанции учета

Набор сообщений «Квитанция статуса прибора учета» («Meter Status Receipt») является обязательным для EVCC/SECC, если он поддерживает набор сообщений «Зарядка переменным током с PnC» или «Зарядка постоянным током с PnC». Несмотря на то что реализация набора сообщений «Meter Status Receipt» является обязательной для обеспечения совместимости, применение данного набора сообщений факультативно.

[V2G2-406] Если EVCC выбрал набор сообщений «Зарядка переменным током с PnC», описанный в 8.6.3.1, и SECC требуется квитанция об учете от EVCC, то SECC должен послать в сообщении ChargingStatusRes параметр ReceiptRequired со значением TRUE.

[V2G2-407] Если EVCC выбрал набор сообщений «Зарядка переменным током с PnC», описанный в 8.6.3.1, и SECC не требуется какое-либо подтверждение приема информации об учете от EVCC, то SECC должен установить в состояние FALSE параметр ReceiptRequired в сообщении ChargingStatusRes.

[V2G2-408] Если EVCC выбрал набор сообщений «Зарядка переменным током с PnC», описанный в 8.6.3.1, то он должен использовать набор сообщений Metering Receipt, а SECC устанавливать в состояние TRUE параметр ReceiptRequired в сообщении ChargingStatusRes.

[V2G2-787] Если EVCC выбрал набор сообщений «Зарядка постоянным током с PnC», описанный в 8.6.3.1, и SECC требуется квитанция об учете от EVCC, то SECC должен использовать факультативный параметр ReceiptRequired в сообщении CurrentDemandRes, устанавливая его в состояние TRUE.

[V2G2-788] Если EVCC выбрал набор сообщений «Зарядка постоянным током с PnC», описанный в 8.6.3.1, и SECC не требуется какое-либо подтверждение приема информации об учете от EVCC, то SECC должен использовать факультативный параметр ReceiptRequired в сообщении CurrentDemandRes, устанавливая его в состояние FALSE.

[V2G2-789] Если EVCC выбрал набор сообщений «Зарядка постоянным током с PnC», описанный в 8.6.3.1, то он должен использовать набор сообщений Metering Receipt, а SECC включать параметр ReceiptRequired со значением TRUE в сообщение CurrentDemandRes.

[V2G2-691] Работая в идентификационном режиме EIM, SECC всегда должен устанавливать параметр ReceiptRequire в состояние FALSE при отправлении сообщения ChargingStatusRes.

Примечание — Для EV может быть важно проверять количество полученной электроэнергии. Если EV найдет расхождение между полученным количеством электроэнергии и количеством электроэнергии, которое показывает EVSE, оно может остановить процесс зарядки и сделать запись соответствующего кода ошибки для последующего анализа (выявления «врезок в линию»).

8.6.3.3 Установка сертификата

[V2G2-410] Если SECC предлагает сервис по установке сертификата в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Установка сертификата».

[V2G2-411] Если EVCC намерен использовать услуги по установке сертификата, предлагаемые SECC в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Установка сертификата».

8.6.3.4 Обновление сертификата

[V2G2-412] Если SECC предлагает сервис по обновлению сертификата в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Обновление сертификата».

[V2G2-413] Если EVCC намерен использовать услуги по обновлению сертификата, предлагаемые SECC в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Обновление сертификата».

8.6.3.5 Набор сообщений дополнительных услуг

[V2G2-414] Если SECC предлагает использование дополнительных услуг в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Value Added Services (VAS)».

[V2G2-415] Если EVCC намерен использовать дополнительные услуги, предлагаемые SECC в параметре ServiceList в сообщении ServiceDiscoveryRes, он должен использовать набор сообщений «Value Added Services (VAS)».

8.6.3.6 Выбор сервисов

В данном пункте установлены зарезервированные диапазоны идентификаторов сервисов (ServiceID) и идентификаторы, которые могут быть использованы с учетом специфики реализации. В таблице 105 приведены идентификаторы ServiceID, определяемые и резервируемые настоящим стандартом. Дополнительно приведены наборы параметров для обслуживания сертификатов и услуг по доступу в Интернет, включая соответствующие идентификаторы наборов параметров ParameterSetID (см. таблицы 106 и 107). В данном пункте также приведены требования, определяющие использование ServiceDiscoveryReq, ServiceDiscoveryRes, ServiceDetailsReq, ServiceDetailsRes, PaymentServiceSelectionReq и PaymentServiceSelectionRes.

Примечание 1 — В J.1 (приложение J) дан пример, как выбирать набор параметров для услуги по доступу в Интернет, используя определения, приведенные в данном пункте.

[V2G2-416] EVCC и SECC должны использовать идентификаторы услуг с 1 по 4, как определено настоящим стандартом.

[V2G2-417] EVCC и SECC должны соответствовать определениям ServiceID, ServiceName, ServiceCategory, данным в таблице 105.

Таблица 105 — Определение ServiceID, Service Category, Service Name

Идентификатор сервиса ServiceID	Наименование сервиса ServiceName	Категория сервиса ServiceCategory	Описание
0			Зарезервировано ИСО/МЭК
1	AC_DC_Charging	EVCharging	Все зарядные сервисы определяются Supported EnergyTransferMode в 8.5.2.3
2	Certificate	ContractCertificate	Сервис, позволяющий обновить или установить сертификаты контракта
3	InternetAccess	Internet	Сервис для стандартных протоколов, таких как HTTP, HTTPs, FTP и т. д.
4	UseCaseInformation	EVSEInformation	Сервис, позволяющий обмениваться специфичной для случая использования информацией об EVSE
5—60 000			Зарезервировано ИСО/МЭК
60 001—65 535			Зарезервировано для реализации конкретного использования

[V2G2-418] ServiceDiscoveryRes должен содержать информацию о предлагаемых услугах, что требует соответствующих ResponseCode, PaymentOptionList, ChargeService и факультативного ServiceList.

[V2G2-419] Требования **[V2G2-420]** и **[V2G2-421]** должны применяться, если предлагается перечень услуг ServiceList.

[V2G2-420] Перечень услуг ServiceList должен содержать список предлагаемых сервисов и для каждого предлагаемого сервиса — информацию о том, может ли данная услуга использоваться бесплатно или потребитель должен за нее платить.

[V2G2-421] Предлагаемый сервис должен определяться его идентификатором ServiceID, предлагаемое значение должно соответствовать таблице 105. Предлагаемый сервис может содержать одно или несколько дополнительных сведений, таких как как ServiceName, ServiceCategory.

[V2G2-422] EVCC должен запрашивать ServiceDetails до использования одного или нескольких сервисов, предлагаемых в перечне ServiceList запроса ServiceDiscoveryRes.

[V2G2-424] EVCC должен указать идентификатор ServiceID, для которого запрашиваются детали сервиса. Идентификатор ServiceID должен использоваться в соответствии с таблицей 105.

[V2G2-425] SECC должен направлять отрицательный код ответа ResponseCode «FAILED_ ServiceIDInvalid», если EVCC указал идентификатор ServiceID, не указанный ранее в запросе ServiceDetailsReq.

[V2G2-426] SECC должен направлять в ответе перечень ServiceParameterList, содержащий подробную информацию о запрошенном идентификаторе ServiceID.

[V2G2-427] ServiceParameterList должен содержать идентификатор набора параметров ParameterSetID и детали, относящиеся к предлагаемым параметрам.

[V2G2-428] ServiceParameterList должен соответствовать таблице 106, если запрашиваются реквизиты сервиса для ServiceID 2 «Certificate».

Таблица 106 — ServiceParameterList для сервиса сертификатов

ParameterSetID (unsigned short)	ParameterName = Service	Описание
0		Зарезервировано ИСО/МЭК
1	stringValue = Установка	Сервис по установке сертификата контракта в EVCC в соответствии с 8.4.3.11
2	stringValue = Обновление	Сервис по обновлению сертификата контракта на EVCC в соответствии с 8.4.3.10
4—60 000		Зарезервировано ИСО/МЭК
60 001—65 535		Реализация конкретного использования

[V2G2-429] ServiceParameterList должен соответствовать таблице 107, если запрашиваются реквизиты сервиса для ServiceID 3 «InternetAccess».

Таблица 107 — ServiceParameterList для сервиса доступа к Интернету

ParameterSet ID (unsigned short)	ParameterName = Protocol	ParameterName = Port	Описание
0			Зарезервировано ИСО/МЭК
1	stringValue = ftp	intValue = 20	Сервис для использования доступа в Интернет с помощью FTP-протокола через порт 20
2	stringValue = ftp	intValue = 21	Сервис для использования доступа в Интернет с помощью FTP-протокола через порт 21
3	stringValue = http	intValue = 80	Сервис для использования доступа в Интернет с помощью HTTP-протокола через порт 80
4	stringValue = https	intValue = 443	Сервис для использования доступа в Интернет с помощью HTTPS-протокола через порт 443
5— 65 535	Имя сервиса в соответствии с *[50]*	Номер порта в соответствии с *[50]*	Дополнительные комбинации портов протоколов, которые поддерживаются SECC для доступа в Интернет

[V2G2-430] Если SECC поддерживает дополнительный протокол/комбинации портов, помимо указанных в таблице 107, то он должен использовать имена сервисов и присвоенные номера портов в соответствии с реестром сервисов и портов *[50]* (т. е. имя сервиса, определенное в *[50]*, передается как «Protocol» и номер порта, определенный в *[50]*, передается как «Port»).

Примечание 2 — Предполагается, что в случае, если имя сервиса и комбинация номеров портов, определенные в *[50]*, применяются для обоих транспортных протоколов, TCP и UDP, SECC поддерживает соединения с TCP или с UDP или с обоими для соответствующей комбинации.

[V2G2-431] PaymentServiceSelectionReq должен содержать список выбранных сервисов.

[V2G2-432] Каждый выбранный сервис должен быть определен в соответствии с ServiceID и ParameterSetID, которые перед этим получены SECC, используя ServiceDiscovery и ServiceDetail Message Set.

[V2G2-433] SECC должен ответить отрицательным ResponseCode «FAILED_ ServiceSelectionInvalid», если EVCC выдал не полученную перед этим пару ServiceID, ParameterSetID в PaymentServiceSelectionReq.

[V2G2-774] Если EVCC и SECC приняли VAS «Internet Access» (Service ID 3, AuthorizationRes с ResponseCode = OK & EVSEProcessing = Finished), SECC должен предоставить VAS для всего сеанса зарядки.

8.7 Хронирование связи V2G

8.7.1 Обзор

В данном подразделе описаны хронирование и обработка ошибок для сеанса связи V2G. Обработка ошибок основывается на таймерах, позволяющих EVCC и SECC осуществлять текущий контроль обмена сообщениями V2G. Для выявления потерянных или несвоевременно доставленных сообщений EVCC и SECC используют предварительно заданные значения тайм-аута как критерии ошибки. Каждый раз, когда таймер равен или больше чем соответствующий тайм-аут, осуществляется соответствующая обработка ошибки.

Таймер подсчитывает временной интервал между моментом времени, когда он был сброшен и вновь запущен. Мониторинг обмена сообщениями V2G основан на двух категориях таймеров:

- таймер сообщений: осуществляет мониторинг обмена парой сообщений запрос-ответ;
- таймер последовательностей: осуществляет мониторинг обмена несколькими парами запрос-ответ.

С целью обработки ошибок при установлении сеанса связи V2G EVCC следит за временем между подключением и получением SessionSetupRes и PowerDeliveryRes соответственно. Это позволяет EVCC принять решение об успешном или неудачном сеансе зарядки по истечении заданных тайм-аутов.

Мониторинг сеанса связи V2G основан на следующих концепциях хронирования:

- хронирование сообщений: осуществляет мониторинг времени между запросом и ответом пары сообщений запрос-ответ. Это позволяет, например, приложению EVCC инициировать обработку ошибок, если ответ не отправлен;
- хронирование последовательности сообщений: осуществляет мониторинг временного согласования последующих пар сообщений запрос-ответ. Это позволяет, например, приложению SECC инициировать обработку ошибок, если ожидаемое следующее сообщение-запрос не было отправлено;
- текущее хронирование: осуществляет мониторинг времени в случае повторяющейся отправки пары запрос-ответ на основании параметра EVSEProcessing, равного «Ongoing». Это позволяет, например, EVCC инициировать обработку ошибок в случае, когда SECC превышает время обработки.

Таймеры сравниваются с предопределенными значениями времени в качестве критерия принятия решения. EVCC и SECC различают две категории:

- тайм-аут: если заданное время превышено, инициируется соответствующая обработка ошибок;
- время исполнения: если заданное время превышено, требование по исполнению не выполнено.

Примечание — Несмотря на то что превышение времени тайм-аута всегда вызывает обработку ошибок, превышение времени исполнения не обязательно влечет за собой обработку ошибок, если требованиями не определено иное. В зависимости от поведения системы (например, времени передачи) ошибка не может возникнуть, если соответствующий коммуникационный партнер не обнаруживает тайм-аута, но вероятность для возникновения тайм-аута при этом высокая.

8.7.2 Последовательность сообщений и сеанс связи

8.7.2.1 Определения

Таймеры сообщений, таймеры последовательности сообщений, тайм-ауты и времена исполнения определены для EVCC и SECC отдельно. Они приведены в таблице 108. Тайм-аут и времена исполнения параметризированы для сообщений отдельно с целью охарактеризовать разные времена обработки. В таблице 109 приведены значения для каждого сообщения V2G.

Таблица 108 — Таймеры, тайм-ауты, времена исполнения, применимые к EVCC и SECC

Имя	Тип	Применимо к	
		EVCC	SECC
V2G_EVCC_Msg_Timer	Таймер сообщений EVCC	x	
V2G_SECC_Msg_Timer	Таймер сообщений SECC		x
V2G_EVCC_Sequence_Timer	Таймер последовательности сообщений EVCC	x	
V2G_SECC_Sequence_Timer	Таймер последовательности сообщений SECC		x
V2G_EVCC_Ongoing_Timer	Таймер текущего времени EVCC	x	
V2G_SECC_Ongoing_Timer	Таймер текущего времени SECC		x
V2G_EVCC_Msg_Timeout (MessageType)	Тайм-аут для таймера сообщений. Значение определяется в зависимости от параметра MessageType в соответствии с таблицей 109	x	
V2G_SECC_Msg_Performance_Time (MessageType)	Время исполнения для таймера сообщений. Значение определяется в зависимости от параметра MessageType в соответствии с таблицей 109		x
V2G_EVCC_Sequence_Performance_Time	Время исполнения для таймера последовательности сообщений в соответствии с таблицей 109	x	
V2G_ SECC_Sequence_Timeout	Тайм-аут для таймера последовательности сообщений в соответствии с таблицей 109		x
V2G_EVCC_Ongoing_Timeout	Тайм-аут для таймера настоящего времени	x	
V2G_ SECC_Ongoing_Performance_Time	Время исполнения для таймера настоящего времени		x

Таблица 109 — Значения параметров хронирования последовательностей сообщений и сеансов связи EVCC и SECC

Имя	Тип сообщения	Значение [с]
V2G_EVCC_Msg_Timeout (MessageType)	SupportedAppProtocolReq	2
	SessionSetupReq	2
	ServiceDiscoveryReq	2
	ServiceDetailReq	5
	PaymentServiceSelectionReq	2
	PaymentDetailsReq	5

Окончание таблицы 109

Имя	Тип сообщения	Значение [с]
V2G_EVCC_Msg_Timeout (MessageType)	AuthorizationReq	2
	ChargeParameterDiscoveryReq	2
	ChargingStatusReq	2
	MeteringReceiptReq	2
	PowerDeliveryReq	5
	CableCheckReq	2
	PreChargeReq	2
	CurrentDemandReq	0,25
	WeldingDetectionReq	2
	SessionStopReq	2
	CertificateInstallationReq	5
	CertificateUpdateReq	5
V2G_ SECC_Msg_Performance_Time(MessageType)	SupportedAppProtocolRes	1,5
	SessionSetupRes	1,5
	ServiceDiscoveryRes	1,5
	ServiceDetailRes	4,5
	PaymentServiceSelectionRes	1,5
	PaymentDetailsRes	4,5
	AuthorizationRes	1,5
	ChargeParameterDiscoveryRes	1,5
	ChargingStatusRes	1,5
	MeteringReceiptRes	1,5
	PowerDeliveryRes	4,5
	CableCheckRes	1,5
	PreChargeRes	1,5
	CurrentDemandRes	0,025
	WeldingDetectionRes	1,5
	SessionStopRes	1,5
	CertificateInstallationRes	4,5
	CertificateUpdateRes	4,5
V2G_EVCC_Sequence_Performance_Time	(все сообщения)	40
V2G_ SECC_Sequence_Timeout	(все сообщения)	60
V2G_EVCC_Ongoing_Timeout	Сообщения-ответы с параметром EVSEProcessing, равным Ongoing	60
V2G_ SECC_Ongoing_Performance_Time	Сообщения-ответы с параметром EVSEProcessing, равным Ongoing	55

Рисунок 98 иллюстрирует, как таймеры сообщений, таймеры последовательностей сообщений, тайм-ауты и времена исполнения применяются в EVCC и SECC.

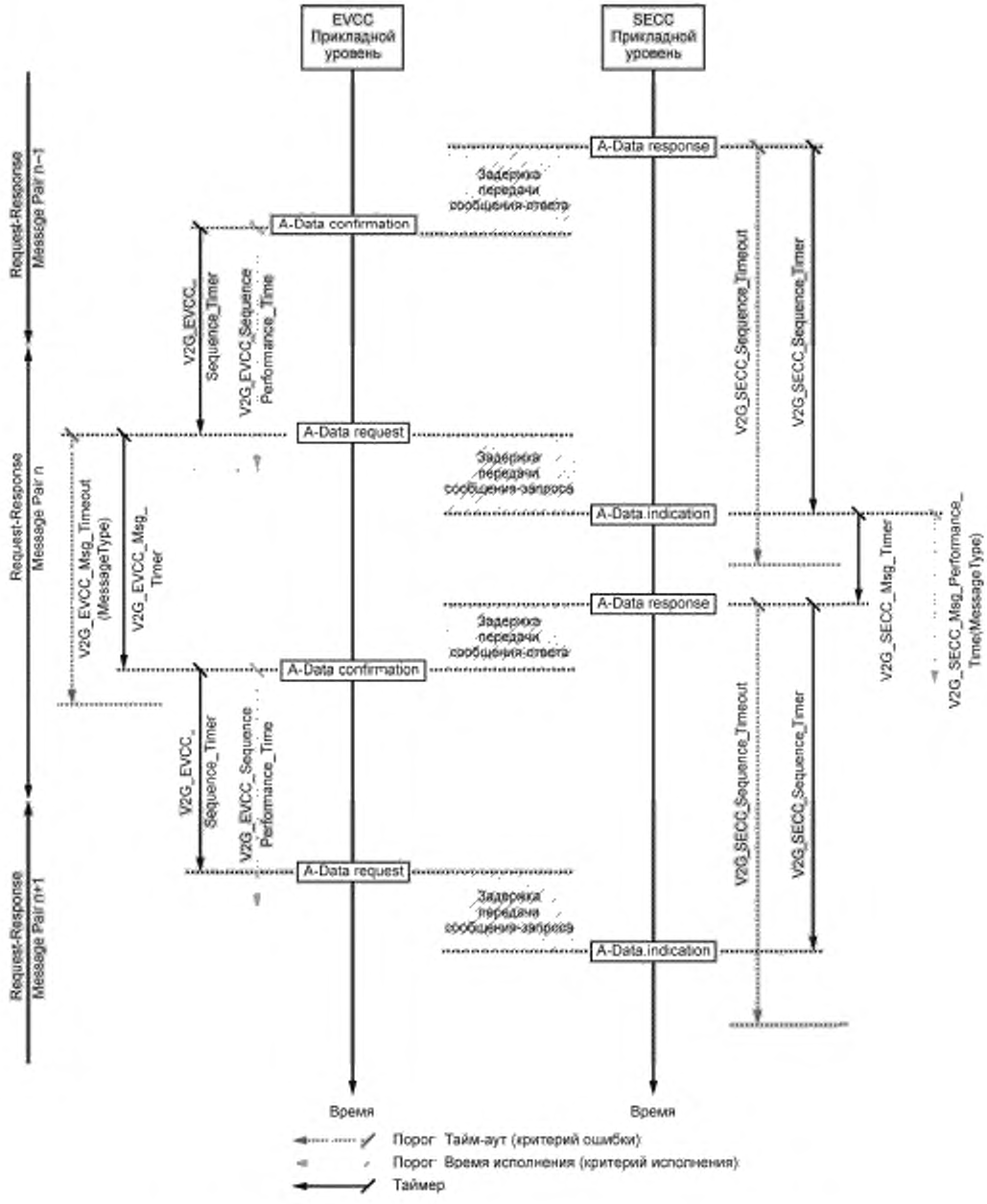


Рисунок 98 — Хронирование последовательности сообщений и сеанса связи

[V2G2-434] EVCC должен выполнять назначенные для EVCC тайм-ауты и времена исполнения, определенные в таблицах 108 и 109.

[V2G2-435] SECC должен выполнять назначенные для SECC тайм-ауты и времена исполнения, определенные в таблицах 108 и 109.

8.7.2.2 Хронирование EVCC пар сообщений запрос-ответ

[V2G2-436] EVCC должен установить тайм-аут V2G_EVCC_Msg_Timeout в зависимости от значения MessageType, как указано в таблице 109, сбросить таймер V2G_EVCC_Msg_Timer и начать мониторинг V2G_EVCC_Msg_Timer после отправки сообщения-запроса.

Примечание 1 — В настоящем стандарте отправление сообщения-запроса описывается действием A-Data.request.

[V2G2-437] EVCC должен ожидать сообщения-ответа, соответствующего переданному перед этим сообщению-запросу.

[V2G2-438] EVCC должен прекратить ожидание сообщения-ответа и прекратить мониторинг V2G_EVCC_Msg_Timer, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout(MessageType), а сообщение-ответ не было получено. Затем он должен применить обработку ошибки, как указано в 8.8.

Примечание 2 — В настоящем стандарте получение сообщения-ответа описывается действием A-Data.confirmation.

[V2G2-439] EVCC должен прекратить ожидание сообщения-ответа и прекратить мониторинг V2G_EVCC_Msg_Timer, когда V2G_EVCC_Msg_Timer меньше, чем V2G_EVCC_Msg_Timeout(MessageType), и он получил сообщение-ответ. Затем он должен обрабатывать сообщение-ответ, как указано в 8.8.

Примечание 3 — В настоящем стандарте получение сообщения-ответа описывается действием A-Data.confirmation.

[V2G2-440] EVCC должен игнорировать любое сообщение, которое не является сообщением-ответом.

8.7.2.3 Хронирование SECC последовательности сообщений запрос-ответ

[V2G2-441] При отправке сообщения-ответа SECC должен установить тайм-аут V2G_SECC_Sequence_Timeout в значение, указанное в таблице 109, сбросить V2G_SECC_Sequence_Timer. Затем начать мониторинг V2G_ SECC_Sequence_Timer.

Примечание 1 — В настоящем стандарте отправление сообщения-ответа описывается действием A-Data.response.

[V2G2-442] SECC должен ожидать сообщения-запроса.

[V2G2-443] SECC должен прекратить ожидание сообщения-запроса и прекратить мониторинг V2G_SECC_Sequence_Timer, когда V2G_SECC_Sequence_Timer больше или равен V2G_ SECC_Sequence_Timeout, а сообщение-запрос не было получено. Затем он должен прекратить сеанс связи V2G.

Примечание 2 — В настоящем стандарте получение сообщения с запросом описывается действием A-Data.indication (A_Msg = «имя сообщения»), которое сигнализирует об успешном получении действительного сообщения-запроса на сообщение V2G, отправленного A_Msg, где «действительное сообщение» означает, что все обязательные элементы заполнены и что оно может быть преобразовано из последовательной формы в параллельную.

[V2G2-444] SECC должен прекратить ожидание сообщения-запроса и прекратить мониторинг V2G_SECC_Sequence_Timer, когда V2G_SECC_Sequence_Timer меньше, чем V2G_SECC_Sequence_Timeout, и он получил сообщение-запрос. Затем он должен обрабатывать сообщение-ответ, как указано в 8.8.

Примечание 3 — В настоящем стандарте получение сообщения-запроса описывается действием A-Data.indication.

[V2G2-445] SECC должен игнорировать любое сообщение, которое не является действительным сообщением-запросом.

8.7.3 Установление сеанса и готовности к зарядке

8.7.3.1 Определения

Временные параметры, применимые к установлению коммуникационной сессии и готовности к зарядке, определяемые настоящим стандартом, представлены в таблице 110. В таблице 111 приведены значения соответствующих времен исполнения и тайм-аутов.

Таблица 110 — Параметры хронирования установления сеанса связи V2G EVCC и SECC

Имя параметра	Определение	Реализация	
		EVCC	SECC
V2G_EVCC_CommunicationSetup_Timer	Таймер установления связи в EVCC	x	
V2G_SECC_CommunicationSetup_Timer	Таймер установления связи в SECC		x
V2G_EVCC_CableCheck_Timer	Таймер проверки кабеля в EVCC	x	
V2G_SECC_CableCheck_Timer	Таймер проверки кабеля в SECC		x
V2G_EVCC_PreCharge_Timer	Таймер предзарядки в EVCC	x	
V2G_SECC_PreCharge_Timer	Таймер предзарядки в SECC		x
V2G_EVCC_CommunicationSetup_Timeout	Тайм-аут для таймера установления связи в соответствии с таблицей 111	x	
V2G_SECC_CommunicationSetup_Performance_Time	Время исполнения для таймера установления связи в соответствии с таблицей 111		x
V2G_EVCC_CableCheck_Timeout	Тайм-аут для таймера проверки кабеля в соответствии с таблицей 111	x	
V2G_SECC_CableCheck_Performance_Time	Время исполнения для таймера проверки кабеля в соответствии с таблицей 111		x
V2G_EVCC_PreCharge_Timeout	Тайм-аут для таймера предзарядки в соответствии с таблицей 111	x	
V2G_SECC_PreCharge_Performance_Time	Время исполнения для таймера предзарядки в соответствии с таблицей 111		x

[V2G2-605] EVCC и SECC должны реализовывать значения параметров хронирования, которые определены в таблице 111.

Таблица 111 — Значения параметров хронирования последовательностей сообщений и сеансов связи EVCC и SECC

Имя параметра	Значение [с]	Реализация	
		EVCC	SECC
V2G_SECC_CommunicationSetup_Performance_Time	18		x
V2G_EVCC_CommunicationSetup_Timeout	20	x	
V2G_SECC_CableCheck_Performance_Time	38		x
V2G_EVCC_CableCheck_Timeout	40	x	
V2G_SECC_PreCharge_Performance_Time	5		x
V2G_EVCC_PreCharge_Timeout	7	x	

Рисунок 99 иллюстрирует, как применяются параметры хронирования, определенные в таблице 110.

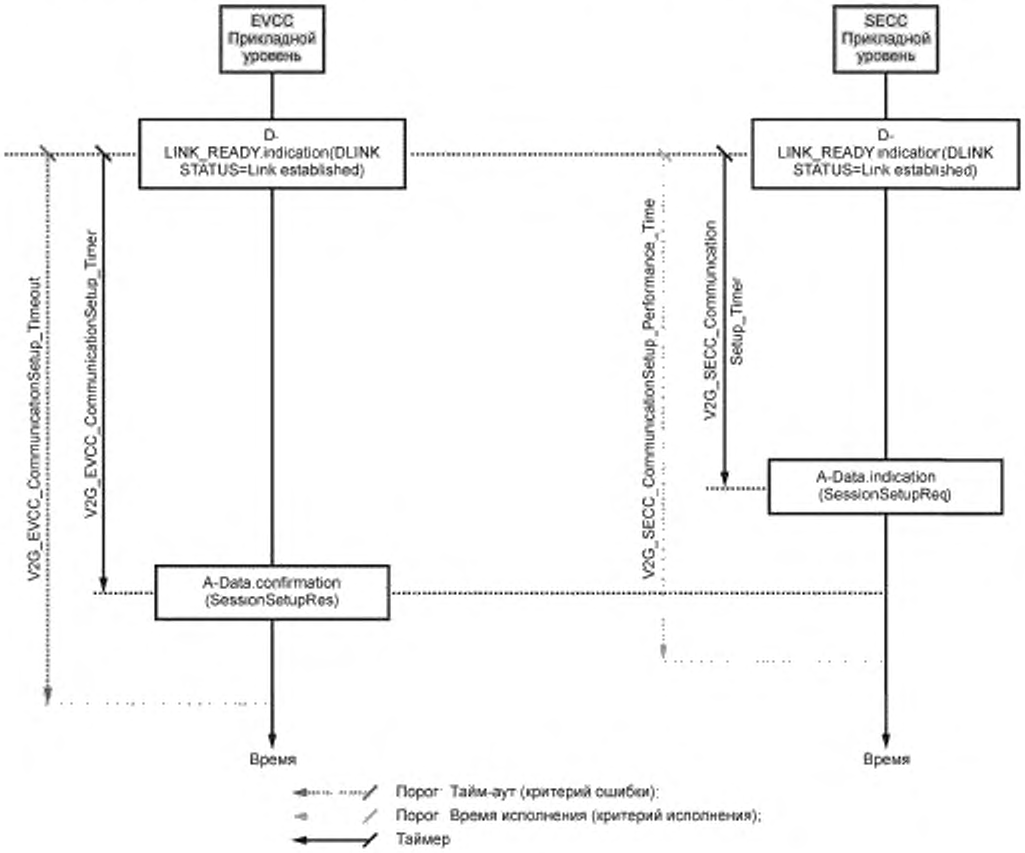


Рисунок 99 — Хронирование установления связи

8.7.3.2 Хронирование EVCC для установления сеанса связи

[V2G2-446] EVCC должен установить тайм-аут V2G_EVCC_CommunicationSetup_Timeout в значение, указанное в таблице 111, сбросить V2G_EVCC_CommunicationSetup_Timer и начать мониторинг V2G_EVCC_CommunicationSetup_Timer, когда будет индицировано успешное установление канала обмена данными [D-LINK_READY. indication(DLINKSTATUS=Link established)].

[V2G2-447] EVCC должен ожидать сообщение SessionSetupRes.

[V2G2-448] EVCC должен прекратить ожидание сообщения SessionSetupRes и прекратить мониторинг V2G_EVCC_CommunicationSetup_Timer, когда V2G_EVCC_CommunicationSetup_ Timer больше или равен V2G_EVCC_CommunicationSetup_Timeout, а сообщение SessionSetupRes не было получено. Затем он должен прекратить сеанс связи V2G.

Примечание 1 — В настоящем стандарте получение сообщения-ответа «SessionSetupRes» описывается действием A-Data.confirmation (SessionSetupRes).

[V2G2-449] EVCC должен прекратить ожидание сообщения SessionSetupRes и прекратить мониторинг V2G_EVCC_CommunicationSetup_Timer, когда V2G_EVCC_CommunicationSetup_ Timer меньше, чем V2G_EVCC_CommunicationSetup_Timeout, и сообщение SessionSetupRes было получено. Затем он должен обработать сообщение-ответ, как указано в 8.8.

Примечание 2 — В настоящем стандарте получение сообщения-ответа «SessionSetupRes» описывается действием A-Data.confirmation (SessionSetupRes).

8.7.3.3 Хронирование SECC для установления сеанса связи

[V2G2-714] SECC должен установить тайм-аут V2G_SECC_CommunicationSetup_Performance_Time в значение, указанное в таблице 110, сбросить V2G_SECC_CommunicationSetup_Timer и начать мониторинг V2G_SECC_CommunicationSetup_Timer при указании на успешное установление канала обмена данными (D-LINK_READY. indication(DLINKSTATUS=Link established).

[V2G2-715] SECC должен ожидать сообщения SessionSetupReq.

[V2G2-716] SECC должен прекратить ожидание сообщения SessionSetupReq и прекратить мониторинг V2G_SECC_CommunicationSetup_Timer, когда V2G_SECC_CommunicationSetup_Timer больше или равен V2G_SECC_CommunicationSetup_Performance_Time, а сообщение SessionSetupRes не было отправлено. Затем он должен применить **[V2G2-034]**.

Примечание 1 — В настоящем стандарте отправление сообщения-запроса «SessionSetupRes» описывается посредством A-Data.response (SessionSetupRes).

8.7.3.4 Хронирование EVCC для параметра EVSEProcessing

[V2G2-710] Если EVCC получает сообщение-ответ V2G с параметром EVSEProcessing, равным «Ongoing», в первый раз, то в сообщении-ответе он должен запустить таймер V2G_EVCC_Ongoing_Timer и ожидать параметра EVSEProcessing, равного «Finished».

[V2G2-711] Если применяется **[V2G2-710]**, то EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Ongoing_Timer больше или равен V2G_EVCC_Ongoing_Timeout и параметр EVSEProcessing, равный «Finished», был получен.

8.7.3.5 Хронирование SECC для параметра EVSEProcessing

[V2G2-712] Если SECC отправляет сообщение-ответ V2G с параметром EVSEProcessing, равным «Ongoing», в первый раз, то в сообщении-ответе он должен запустить таймер V2G_SECC_Ongoing_Timer.

[V2G2-713] Если применяется **[V2G2-712]**, то SECC должен отправить ResponseCode, равный «FAILED», когда V2G_SECC_Ongoing_Timer больше или равен V2G_SECC_Ongoing_Performance_Time, а параметр EVSEProcessing, равный «Finished», не был отправлен. SECC должен остановить сеанс связи V2G.

8.7.3.6 Хронирование EVCC для проверки кабеля

[V2G2-700] EVCC должен установить тайм-аут V2G_EVCC_CableCheck_Timeout в значение, указанное в таблице 111, сбросить V2G_EVCC_CableCheck_Timer и начать мониторинг V2G_EVCC_CableCheck_Timer при отправлении сообщения CableCheckReq в первый раз во время сеанса зарядки.

Примечание 1 — В настоящем стандарте отправление сообщения-запроса описывается посредством A-Data.request.

[V2G2-701] EVCC должен ожидать завершения проверки кабеля EVSE, на которое указывает получение CableCheckRes с ResponseCode, равным «OK», и EVSEProcessing, равное «Finished».

Примечание 2 — В настоящем стандарте получение сообщения-запроса описывается посредством A-Data.confirmation.

[V2G2-702] EVCC должен прекратить ожидание завершения проверки кабеля EVSE и прекратить мониторинг V2G_EVCC_CableCheck_Timer, если V2G_EVCC_CableCheck_Timer больше или равен V2G_EVCC_CableCheck_Timeout. Затем он должен применить обработку ошибок, как указано в 8.8.

[V2G2-703] EVCC должен прекратить ожидание завершения проверки кабеля EVSE и прекратить мониторинг V2G_EVCC_CableCheck_Timer, если V2G_EVCC_CableCheck_Timer меньше чем V2G_EVCC_CableCheck_Timeout и было получено сообщение CableCheckRes с ResponseCode, равным «OK», и EVSEProcessing, равное «Finished». Затем он должен обработать сообщение-ответ, как описано в 8.8.

Примечание 3 — В настоящем стандарте получение сообщения-запроса описывается посредством A-Data.confirmation.

8.7.3.7 Хронирование EVCC для предзарядки

[V2G2-704] Изначально во время сеанса зарядки EVCC должен установить тайм-аут V2G_EVCC_PreCharge_Timeout в значение, указанное в таблице 111, сбросить V2G_EVCC_PreCharge_Timer и начать мониторинг V2G_EVCC_PreCharge_Timer при отправлении сообщения PreChargeReq.

Примечание — В настоящем стандарте отправление сообщения-запроса описывается посредством A-Data.request.

[V2G2-705] EVCC должен дождаться завершения предварительной зарядки, о чем информирует EV, которое определяет, что напряжение EVSE соответствует напряжению аккумуляторной батареи.

[V2G2-706] EVCC должен прекратить ожидание завершения предзарядки и прекратить мониторинг V2G_EVCC_PreCharge_Timer, если V2G_EVCC_PreCharge_Timer больше или равен V2G_EVCC_PreCharge_Timeout. Затем он должен применить обработку ошибки, как описано в 8.8.

[V2G2-707] EVCC должен прекратить мониторинг V2G_EVCC_PreCharge_Timer, если V2G_EVCC_PreCharge_Timer меньше, чем V2G_EVCC_PreCharge_Timeout, а предзарядка была завершена. Затем EVCC должен сформировать сообщение-ответ, как описано в 8.8.

8.7.4 Синхронизация сообщений V2G с сигналами, описанными в *ГОСТ Р МЭК 61851-1*

8.7.4.1 Обзор

Управление зарядкой по *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и в соответствии с настоящим стандартом расширяет возможность зарядки с обменом сигналами по *ГОСТ Р МЭК 61851-1*. Для этого обмен сообщениями на прикладном уровне синхронизируется с состояниями контрольного управления, определенными в *ГОСТ Р МЭК 61851-1*.

Обмен сообщениями на основе *[1], ГОСТ Р 58122 (ИСО 15118-1*:2013) и настоящего стандарта способен обеспечить управление процессом зарядки переменным и постоянным током во время всего сеанса зарядки (от начала до конца) при коэффициенте заполнения 5 %.

В случае зарядки переменным током в соответствии с *[1]* это также позволяет начать зарядку на основе *ГОСТ Р МЭК 61851-1* (базовая зарядка, BC), а затем переключиться на управление зарядкой на основе *[1], ГОСТ Р 58122 (ИСО 15118-1*) и настоящего стандарта (управление связью по протоколу высокого уровня, HLC-C).

В данных пунктах термины и определения в требованиях применяются, как определено в *[1]* и *ГОСТ Р МЭК 61851-1*.

На рисунке 100 показан пример зарядки переменным и постоянным током на основе BC и HLC-C с привязкой к этапам установления канала связи, установления V2G и к циклу зарядки во время коммуникационной сессии V2G. На рисунке также показан пример наиболее важных условий входа и выхода из этапов. В случае переменного тока может быть применен этап BC. В общем случае BC перед HLC-C требует номинального коэффициента заполнения вместо 5 %.

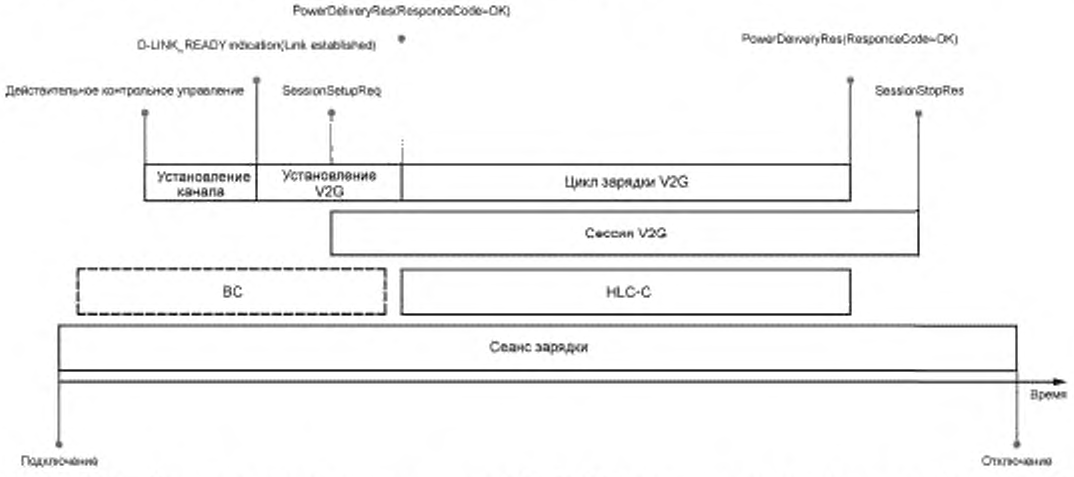


Рисунок 100 — Пример зарядки переменным током с BC и HLC-C по отношению к коммуникационной сессии V2G

Цикл зарядки V2G определяется как обмен сообщениями V2G с целью управления процессом зарядки при нормальном функционировании. Этап зарядки по *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящему стандарту определяется как зарядка под управлением высокого уровня (HLC-C).

Входное и выходное условия цикла зарядки V2G следующие:

- входное условие: сообщение PowerDeliveryRes с параметром ResponseCode, равным OK, после сообщения PowerDeliveryReq с ChargeProgress, равным «Start»;
- выходное условие: сообщение PowerDeliveryRes с параметром ResponseCode, равным OK, после сообщения PowerDeliveryReq с ChargeProgress, равным «Stop».

Кроме требований к парам сообщений запрос-ответ и последовательностям сообщений запрос-ответ, описанных в 8.8.4, EVCC и SECC должны соответствовать требованиям, определенным в 8.7.4.2—8.7.4.4.

8.7.4.2 Общие требования

К EV применяются следующие требования:

[V2G2-836] EVCC должен использовать сообщение PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Start», чтобы начать зарядку на базе HLC-C.

[V2G2-837] Если применяется **[V2G2-836]** и сообщение PowerDeliveryRes получено с ResponseCode, установленным в OK, то EV должно применять предельные параметры зарядки в соответствии с обменом сообщениями V2G.

[V2G2-838] EV должно прекращать зарядку до отправления сообщения PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop».

[V2G2-839] EVCC должен использовать сообщение PowerDeliveryReq с параметром ChargeProgress, установленным в «Stop», для прекращения зарядки на базе HLC-C.

[V2G2-840] EVCC должен использовать сообщение PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiation», с целью инициирования обмена сообщениями повторного согласования зарядки.

[V2G2-841] EVCC не должен посылать сообщение PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiation», до отправления PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Start» (см. **[V2G2-837]**).

[V2G2-842] Если применяется **[V2G2-841]**, EVCC должен устанавливать параметр ChargeProgress в значение «Start» в последующем сообщении PowerDeliveryReq для применения согласованных параметров зарядки после повторного согласования.

[V2G2-843] Реализованные на основе *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящего стандарта EV должны соответствовать *ГОСТ Р МЭК 61851-1*.

[V2G2-844] На протяжении процесса HLC-C EV должно применять все ограничения зарядки согласно *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящему стандарту и дополнительно — ограничения, определенные в *ГОСТ Р МЭК 61851-1*.

Примечание 1 — Только для зарядки постоянным током применяются ограничения, описанные в *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящем стандарте.

[V2G2-845] Если выбирается последовательность сообщений «EIM charging AC» или «EIM charging DC» и параметр EVSEProcessing устанавливается в значение «Ongoing_ WaitingForCustomerInteraction» в AuthorizationRes, EV должно повторно послать сообщение AuthorizationReq.

[V2G2-731] Если применяется **[V2G2-508]** или **[V2G2-728]**, EVCC может установить новую или восстановить начатую сессию связи V2G путем применения **[V2G2-014]**.

[V2G2-737] Если EVCC обнаруживает состояния E или F контрольного управления, он должен остановить сессию связи V2G и продолжить с **[V2G2-728]**.

Следующие требования предъявляются к EVSE:

[V2G2-850] Реализованные на основе *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящего стандарта EVSE должны соответствовать *ГОСТ Р МЭК 61851-1*.

[V2G2-854] Если идентификационный режим EIM был выбран параметром SelectedPaymentOption, равным «External Payment», в сообщении ServicePaymentSelectReq и отсутствует положительная информация EIM, то SECC должен установить в значение «Ongoing_ WaitingForCustomerInteraction» параметр EVSEProcessing в AuthorizationRes.

[V2G2-855] Если идентификационный режим PnC был выбран параметром SelectedPaymentOption, равным «Contract», в сообщении ServicePaymentSelectReq и отсутствует положительная информация PnC, то SECC должен установить в значение «Ongoing» параметр EVSEProcessing в AuthorizationRes.

[V2G2-856] Если имеется положительная информация авторизации SECC, то он должен установить в значение «Finished» параметр EVSEProcessing в AuthorizationRes.

Примечание 2 — **[V2G2-856]** применяет одинаковые действия для положительной авторизации при идентификационном режиме EIM и PnC.

[V2G2-733] Если применяется **[V2G2-571], [V2G2-539]** или **[V2G2-729]**, то SECC может разрешить установку новой или возобновление начатой сессии связи V2G путем применения **[V2G2-027]**.

8.7.4.3 Специальные требования для переменного тока

К EV применяются следующие требования:

Перед входом в режим HLC-C с 5 % PWM EV должно выдать сигнал состояния B контрольного управления.

[V2G2-930] EV должно определять коэффициент заполнения PWM в 5 % или номинальный коэффициент заполнения перед отправлением сообщения ChargeParameterDiscoveryReq.

[V2G2-846] В случае, если коэффициент заполнения PWM имеет значение 5 % или номинальное значение без ВС, EV должно выдать сигнал состояния B контрольного управления перед отправкой первого сообщения PowerDeliveryReq с параметром ChargeProgress, равным «Start», внутри SessionPowerDeliveryReq.

[V2G2-847] EV должно выдавать сигнал состояния C или D контрольного управления не более чем через 250 мс после отсылки первого сообщения PowerDeliveryReq с параметром ChargeProgress, равным «Start», в SessionPowerDeliveryReq сессии связи V2G.

[V2G2-848] EV должно выдать сигнал состояния B контрольного управления не более чем через 250 мс после отсылки сообщения PowerDeliveryReq с параметром ChargeProgress, равным «Stop».

[V2G2-849] Если применяется **[V2G2-847]** и не было обнаружено ошибок, то EV не должно изменять состояние контрольного управления до применения **[V2G2-848]**.

Примечание 1 — В случае пересогласования контакт остается замкнутым, позволяя осуществлять зарядку на базе имеющихся ограничений в течение процесса пересогласования.

К EVSE применяются следующие требования:

EVSE должно выдавать сигнал PWM с коэффициентом заполнения 5 % для сообщения EV, что *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящим стандартом требуется для зарядки. При PWM с коэффициентом заполнения 10—95 % и EIM *[1], ГОСТ Р 58122 (ИСО 15118-1:2013)* и настоящий стандарт могут быть применены одновременно с *ГОСТ Р МЭК 61851-1*.

[V2G2-931] EVSE должно выдавать сигнал PWM с коэффициентом заполнения 5 % или с номинальным коэффициентом заполнения после отсылки сообщения AuthorizationRes.

[V2G2-851] Если передача электроэнергии невозможна, то EVSE должно выдавать PWM с коэффициентом заполнения 100 %.

[V2G2-852] Если информация о положительной авторизации от EIM не получена, то оно должно выдавать сигнал PWM с коэффициентом заполнения, равным 5 %.

[V2G2-853] Если EVSE получило информацию о положительной авторизации от EIM, то EVSE должно выдавать PWM с номинальным коэффициентом заполнения.

SECC ожидает положительную авторизацию и возможности заряда перед активацией цикла зарядки.

[V2G2-857] Если применяется **[V2G2-856]** и коэффициент заполнения сигнала PWM, выдаваемого EVSE, равен 5 %, то EVSE не должно изменять значение коэффициента заполнения до окончания сессии связи V2G.

Примечание 2 — Если EVSE не может обеспечить передачу электроэнергии (EVSE выдает сигнал PWM с коэффициентом заполнения, равным 100 %), то EVSE может сообщить в параметре SASchedule сообщения ChargeParameterDiscoveryRes время, когда заряд будет доступен. В этом случае EV имеет возможность сделать паузу и передать сообщение PowerDeliveryReq с параметром ChargeProgress, равным «Stop».

Контактор EVSE должен быть замкнут до отправления сообщения PowerDeliverRes.

[V2G2-858] После получения PowerDeliveryReq с параметром ChargeProgress, равным «Start», SECC должен осуществлять мониторинг состояния контактора и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-859] В случае зарядки на базе коммуникации высокого уровня SECC не должен замыкать контактор до получения PowerDeliveryReq(ChargeProgress = Start).

[V2G2-860] Если не обнаруживаются ошибки, то SECC должен замыкать контактор не позже чем через 3 с после обнаружения состояний C или D управления.

[V2G2-861] SECC должен отвечать сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если SECC определяет, что контактор замкнут.

[V2G2-862] SECC должен отвечать сообщением PowerDeliveryRes, содержащим «ResponseCode = FAILED_ContactorError», если V2G_SECC_Msg_Timer больше или равен V2G_SECC_Msg_Performance в «PowerDelivery» в соответствии с таблицей 109 и SECC определяет, что контактор разомкнут.

[V2G2-863] После получения PowerDeliveryReq с параметром ChargeProgress, равным «Stop», SECC должен осуществлять мониторинг состояния контактора и запускать V2G_SECC_Msg_Performance_Timer.

[V2G2-864] SECC должен отвечать сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если SECC определяет, что контактор разомкнут.

[V2G2-865] SECC должен отвечать сообщением PowerDeliveryRes, содержащим «ResponseCode = FAILED_ContactorError», если V2G_SECC_Msg_Timer больше или равен V2G_SECC_Msg_Performance в «PowerDelivery» в соответствии с таблицей 109 и SECC определяет, что контактор замкнут.

[V2G2-866] Если применяется **[V2G2-862]** или **[V2G2-865]**, то SECC должен устанавливать значение коэффициента заполнения сигнала PWM, равное 100 %, не позднее чем через 250 мс после отсылки PowerDeliveryRes.

Примечание 3 — В настоящем стандарте, *ГОСТ Р 58122 (ИСО 15118-1:2013)* и *[1]* не приведено конкретных требований к реализации управления контактором в EVSE. Управление силовым контактором зависит от архитектуры EVSE.

Примечание 4 — Для переменного тока последний момент в последовательности обмена сообщениями до начала заряда (замыкающий контактор, напряжение на входе EV) — перед отправкой сообщения PowerDeliveryRes SECC. Если состояние C или D управления было сообщено раньше, EVSE может начать передачу электроэнергии тоже раньше. Например, если EVSE и EV поддерживают BC перед HLC-C, EVSE формирует сигнал PWM с коэффициентом заполнения между 10 и 96 %, EV сообщает состояние C или D управления, а EVSE замыкает контактор и подает энергию, как определено *ГОСТ Р МЭК 61851-1*. В любом случае должны соблюдаться временные ограничения, приведенные в *ГОСТ Р МЭК 61851-1*.

8.7.4.4 Специальные требования для постоянного тока

Следующие требования предъявляются к EV:

[V2G2-912] EVCC должен изменить состояние управления на C или D, как описано в *ГОСТ Р МЭК 61851-1*, после получения сообщения ChargeParameterDiscoveryRes и до отправления сообщения CableCheckReq.

[V2G2-913] После отправления сообщения PowerDeliveryReq с ChargeProgress, равным «STOP», и получения соответствующего сообщения PowerDeliveryRes EVCC до отправления следующего сообщения-запроса должен изменить состояние управления на B, как описано в *ГОСТ Р МЭК 61851-1*.

[V2G2-914] Если применяется **[V2G2-912]** и не было обнаружено ошибок, то EV не должно изменять состояние управления до применения **[V2G2-913]**.

К EVSE применяются следующие требования:

[V2G2-915] EVSE должны применять коэффициент заполнения 5 % от начала установления канала связи до окончания сессии V2G.

SECC ожидает положительную авторизацию и наличие возможности передачи электроэнергии до начала цикла зарядки.

[V2G2-916] После получения сообщения CableCheckReq SECC должен осуществлять мониторинг состояния управления и запускать V2G_SECC_Msg_Performance_Timer.

[V2G2-917] SECC должен отвечать сообщением CableCheckRes, содержащим «ResponseCode = OK», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если SECC определяет состояние C или D управления.

[V2G2-918] SECC должен отвечать сообщением CableCheckRes, содержащим «ResponseCode = FAILED», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если SECC не определяет состояние C или D управления.

[V2G2-919] После отправления сообщения PowerDeliveryRes с параметром ResponseCode, установленным в значение «OK», в ответ на PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop», SECC должен пытаться обнаружить состояние B управления.

[V2G2-920] После получения сообщения WeldingDetectionReq или сообщения SessionStopReq SECC должен ожидать максимального V2G_SECC_Msg_Performance_Time для обнаружения состояния B управления.

[V2G2-921] После получения сообщения WeldingDetectionReq или сообщения SessionStopReq SECC должен отправить соответствующее сообщение-ответ с параметром ResponseCode, установленным в значение «OK», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 103, если было обнаружено состояние B управления.

[V2G2-922] После получения сообщения WeldingDetectionReq или сообщения SessionStopReq SECC должен отправить соответствующее сообщение-ответ с параметром ResponseCode, установленным в значение «FAILED», в течение V2G_SECC_Msg_Performance_Time в соответствии с таблицей 103, если состояние B управления не было обнаружено.

Примечание — Если EVSE не может обеспечить передачу электроэнергии (EVSE формирует сигнал PWM со значением коэффициента заполнения 100 %), то оно может информировать о времени доступности электроэнерии в параметре SASchedule сообщения ChargeParameterDiscoveryRes. В этом случае EV имеет возможность сделать паузу, начав с передачи сообщения PowerDeliveryReq с параметром ChargeProgress, равным «Stop».

8.8 Задание последовательности сообщений и обработка ошибок

8.8.1 Обзор

В течение сессии связи V2G EVCC и SECC обмениваются парами сообщений запрос-ответ на основе предопределенной последовательности. В настоящем стандарте даются ссылки на последовательность сообщений запрос-ответ. Эти последовательности сообщений позволяют обеим сторонам синхронизировать процесс и управлять коммуникацией между двумя объектами V2G.

Основная концепция обработки ошибок в настоящем стандарте базируется на применении таймеров прикладного уровня для последовательностей сообщений запрос-ответ. Это позволяет V2G контролировать на прикладном уровне любую ошибку процесса и ошибку коммуникации после ожидания запланированных реакций до истечения тайм-аута. Таким образом, V2G может принять решение о возобновлении процесса после получения положительного результата до истечения тайм-аута. Если ошибки на уровнях коммуникации, которые ниже прикладного уровня, отсутствуют, приложение имеет возможность прервать коммуникацию TCP в случае наличия ошибок приложения. Прерванное соединение TCP перед SessionStopRes всегда интерпретируется как ошибка EVCC или SECC.

В настоящем стандарте установлены концепции тайм-аута, но не установлены ограничения по отношению к дополнительным механизмам выявления ошибок при условии, что эти механизмы не приводят к несовместимости.

В течение сеанса зарядки EVCC может установить новую сессию связи V2G после возникновения ошибки, как описано в 7.4. В этом случае EVCC и SECC начинают коммуникацию так же, как и при установлении первой сессии V2G.

В общем случае время обработки сообщения-запроса в SECC ограничено требованиями, определенными в 8.7.2.

Если SECC необходимо больше времени для обработки, он может применить параметр EVSEProcessing (установленный в значение Ongoing), чтобы избежать тайм-аута в EVCC путем конкретного указания на то, что обработка еще не закончена и EVCC должен подождать окончания обработки, прежде чем приступить к формированию последовательности сообщений. После этого EVCC

должен периодически повторять сообщение-запрос до тех пор, пока EVSE не сообщит об окончании обработки параметром EVSEProcessing (установленным в значение «Finished»).

8.8.2 Базовые определения для обработки ошибок

Базовая обработка ошибок пары сообщений запрос-ответ и последовательности сообщений запрос-ответ основывается на ResponseCode, включенном в сообщение-ответ от SECC. В зависимости от значения ResponseCode EVCC принимает решение, может ли он продолжать работу со стандартной последовательностью сообщений запрос-ответ или необходимо обрабатывать ошибку.

В настоящем стандарте ResponseCode, как определено в F.6 (приложение F), интерпретируется EVCC следующим образом:

- положительный ответ (OK):

любое значение, начинающееся с «OK» или «OK_», указывает на положительный ответ. Детальная информация может быть предоставлена с помощью OK_<additional info>. Эта информация может быть использована для установления различий в реакции на положительный ответ;

- отрицательный ответ (FAILED):

любое значение, начинающееся с «FAILED» или «FAILED_», указывает на отрицательный ответ. Детальная информация может быть предоставлена с помощью FAILED_<additional info>. Эта информация может быть использована для установления различий в реакции на отрицательный ответ.

Обработка EVCC остальных параметров в сообщении-ответе зависит от параметра ResponseCode. Если ResponseCode содержит значение, начинающееся с OK, указывающее на то, что ошибок выявлено не было, EVCC может обрабатывать другие параметры сообщения-ответа в обязательном порядке или факультативно, как указано в 8.6. Если ResponseCode содержит значение, начинающееся с FAILED, указывающее на то, что была выявлена ошибка, EVCC должен игнорировать другие параметры сообщения-ответа.

Если содержание полученного сообщения-запроса корректно и это сообщение-запрос считается приемлемым при данном текущем состоянии (последовательности состояний) SECC, то ResponseCode устанавливается в значение «OK». Это также верно для сообщений-ответов с EVSEProcessing, установленным в значение Ongoing. Кодом ResponseCode, установленным в значение «OK», SECC указывает, что сообщение-запрос считается приемлемым и корректным, но окончательный ответ не может быть выдан в данный момент, поэтому требуется время для отсылки допустимых значений в сообщении-ответе. Если в сообщении-ответе содержатся допустимые значения, то SECC указывает на это параметром EVSEProcessing (установленным в значение Finished) и EVCC может продолжать последовательность, как планировалось для первого сообщения-запроса перед получением параметра EVSEProvessing (установленного в значение Ongoing). Если SECC устанавливает ResponseCode в значение FAIL, то это указывает на ошибку в процессе или на некорректное сообщение. При этом применяется стандартная обработка ошибок.

8.8.3 Обработка ResponseCode

8.8.3.1 Общие требования

Кроме требований к парам сообщений запрос-ответ и последовательностям сообщений запрос-ответ, описанным в 8.8.4, EVCC и SECC должны соответствовать следующим требованиям.

[V2G2-734] Если ResponseCode содержит значение, начинающееся с OK, то EVCC должен обрабатывать другие параметры сообщения-ответа, как указано в 8.8.

[V2G2-735] Если ResponseCode содержит значение, начинающееся с FAILED, то EVCC должен игнорировать другие параметры сообщения-ответа.

[V2G2-736] Если SECC отправляет ResponseCode, содержащий значение, начинающееся с FAILED, то все обязательные параметры должны быть заполнены произвольно выбранными соответствующими XSD значениями. Кроме того, все значения должны быть выбраны таким образом, чтобы сообщение-ответ имело минимальный размер.

В общем случае каждое сообщение-ответ может содержать два типа значений ResponseCode: «OK» или «FAILED».

[V2G2-457] Если обработка сообщения-запроса была успешной, то сообщение-ответ должно содержать ResponseCode «OK» в атрибуте «ResponseCode». Если затем для конкретной ситуации определяется конкретный положительный «ResponseCode», то необходимо использовать этот ResponseCode.

[V2G2-458] Сообщение-ответ должно содержать ResponseCode «FAILED» в атрибуте «ResponseCode», если обработка сообщения-запроса не была успешной и специальный «ResponseCodeType» не назначен для конкретного случая ошибки.

[V2G2-459] Сообщение-ответ должно содержать ResponseCode «FAILED_SequenceError», если SECC получил непредвиденное сообщение-запрос.

[V2G2-460] Сообщение-ответ должно содержать ResponseCode «FAILED_UnknownSession», если SessionID в любом сообщении-запросе, за исключением SessionSetupReq, не равно значению SessionID, запомненному для активной в настоящий момент сессии связи V2G.

[V2G2-461] Сообщение-ответ должно содержать ResponseCode «FAILED_SignatureError», если проверка защитного элемента в заголовке сообщения не удалась.

[V2G2-462] Сообщение «SessionSetupRes» должно содержать ResponseCode «OK_NewSessionEstablished», если обработка сообщения SessionSetupReq прошла успешно и в сообщении-ответе содержится другой SessionID по сравнению с SessionID в сообщении-запросе.

[V2G2-463] Сообщение «SessionSetupRes» должно содержать ResponseCode «OK_OldSessionJoined», если обработка сообщения SessionSetupReq прошла успешно и в сообщении-запросе содержится такой же SessionID, как и в сообщении-ответе.

[V2G2-464] Сообщение «ServiceDetailRes» должно содержать ResponseCode «FAILED_ServiceIDInvalid», если ServiceID сообщения ServiceDetailReq не был частью предлагаемого ServiceList во время ServiceDiscovery.

[V2G2-465] Сообщение «PaymentServiceSelectionRes» должно содержать ResponseCode «FAILED_PaymentSelectionInvalid», если SelectedPaymentOption, содержащийся в сообщении PaymentServiceSelectionReq, не был частью предлагаемого PaymentOptionList в ServiceDiscoveryRes.

[V2G2-467] Сообщение «PaymentServiceSelectionRes» должно содержать ResponseCode «FAILED_ServiceSelectionInvalid», если SelectedServiceList, содержащийся в сообщении PaymentServiceSelectionReq, содержит ServiceID, который не был в составе предлагаемого ServiceList в ServiceDiscoveryRes.

[V2G2-804] Сообщение «PaymentServiceSelectionRes» должно содержать ResponseCode «FAILED_NoChargeServiceSelected», если SelectedServiceList, содержащийся в сообщении PaymentServiceSelectionReq, не содержит значение ServiceID, равное 1, которое было предложено SECC посредством параметра ChargeService в ServiceDiscoveryRes.

[V2G2-468] Сообщение «CertificateInstallationRes» должно содержать ResponseCode «FAILED_CertificateExpired», если OEMProvisioningCert, содержащийся в сообщении CertificateInstallationReq, не действителен.

[V2G2-469] Сообщение «CertificateInstallationRes» должно содержать ResponseCode «FAILED_NoCertificateAvailable», если новый сертификат не может быть получен от SA в течение времени V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109.

[V2G2-470] Сообщение «CertificateUpdateRes» должно содержать ResponseCode «FAILED_CertChainError», если ContractSignatureCertChain, содержащийся в CertificateInstallationReq, не действителен.

[V2G2-471] Сообщение «CertificateUpdateRes» должно содержать ResponseCode «FAILED_NoCertificateAvailabl», если новый сертификат не может быть получен от вторичного субъекта в течение времени V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109.

[V2G2-472] Сообщение «CertificateUpdateRes» должно содержать ResponseCode FAILED_ContractCanceled, если EMAID, предоставленный EVCC во время CertificateUpdateReq, не считается приемлемым для SA.

[V2G2-473] Сообщение «CertificateUpdateRes» должно содержать ResponseCode «FAILED_CertificateExpired», если сертификат контракта, содержащийся в сообщении CertificateUpdateReq, не действителен.

[V2G2-474] Сообщение «PaymentDetailsRes» должно содержать ResponseCode «FAILED_CertificateExpired», если просрочен сертификат контракта, содержащийся в сообщении PaymentDetailReq в атрибуте ContractSignatureCertChain.

[V2G2-824] Сообщение «PaymetDetailsRes» должно содержать ResponseCode «FAILED_ CertificateExpired», если сертификат контракта, содержащийся в сообщении PaymentDetailsReq в атрибуте ContractSignatureCertChain, еще не действителен.

[V2G2-475] Сообщение «AuthorizationRes» должно содержать ResponseCode «FAILED_ ChallengeInvalid», если ответ на вызов, содержащийся в AuthorizationReq в атрибуте GenChallenge, не действителен по отношению к предоставленному GenChallenge в PaymentDetailsRes.

[V2G2-476] Сообщение «ChargeParameterDiscoveryRes» должно содержать ResponseCode «FAILED_WrongEnergyTransferMode», если содержание атрибута «Requested EnergyTransferMode» в сообщении ChargeParameterDiscoveryReq не действительно или не соответствует содержанию атрибута EVChargeParameter.

[V2G2-477] Сообщение «ChargeParameterDiscoveryRes» должно содержать ResponseCode «FAILED_WrongChargeParameter», если содержание атрибута «EVChargeParameter» в сообщении ChargeParameterDiscoveryReq не действительно, например дан неверный набор параметров, один или несколько параметров не могут быть интерпретированы.

[V2G2-478] Сообщение «PowerDeliveryRes» должно содержать ResponseCode «FAILED_ ChargingProfileInvalid», если содержание атрибута «ChargingProfile» в сообщении PowerDeliveryReq нарушает ограничения по мощности, указанные в «ChargeParameterDiscoveryRes».

[V2G2-479] Сообщение «PowerDeliveryRes» должно содержать ResponseCode «FAILED_ TariffSelectionInvalid», если содержание атрибута «ChargingProfile» в сообщении PowerDeliveryReq содержит SAtupleID, которого не было в составе атрибута «SASchedules», представленного в «ChargeParameterDiscoveryRes».

[V2G2-480] Сообщение «PowerDeliveryRes» должно содержать ResponseCode «FAILED_ PowerDeliveryNotApplied», если EVSE не может передавать электроэнергию.

[V2G2-481] Сообщение «MeteringReceiptRes» должно содержать ResponseCode «FAILED_ MeteringSignatureNotValid», если SECC не может проверить подпись или содержащиеся показания прибора учета не соответствуют показаниям, представленным при «ChargingStatusRes», и при этом SECC требует, чтобы подпись была действительной.

[V2G2-690] Если SECC способен сопоставить пригодность сертификата контракта ContractCertificate с настоящим временем, то SECC должен установить responseCode для PaymentServiceSelectionRes в значение OK_CertificateExpiresSoon при условии, что срок действия полученного сертификата истекает в течение 21 дня.

[V2G2-693] Сообщение AuthorizationRes должно содержать ResponseCode «FAILED_ CertificateRevoked», если SECC сличает ContractCertificate с CRL и ContractCertificate отмечается как отозванный.

[V2G2-694] Сообщение «CertificateInstallationRes» должно содержать ResponseCode «FAILED_ CertificateRevoked», если SECC сличает сервисный сертификат изготовителя с CRL и ContractCertificate отмечается как отозванный.

[V2G2-695] Сообщение «CertificateUpdateRes» должно содержать ResponseCode «FAILED_ CertificateRevoked», если SECC сличает ContractCertificate с CRL и ContractCertificate отмечается как отозванный.

Примечание — В зависимости от реализации могут использоваться коды ResponseCodes, не определенные в этом разделе.

В таблице 112 представлен обзор значений ResponseCode и сообщений.

Таблица 112 — Обзор применения ResponseCode

ResponseCode (Enumeration)	Сообщения квитирования прикладного уровня V2G	Сообщения прикладного уровня V2G												
	supportedAppProtocolRes	SessionSetupRes	ServiceDiscoveryRes	ServiceDetailRes	ServicePaymentSelectionRes	PaymentDetailsRes	AuthorizationRes	ChargeParameterDiscoveryRes	PowerDeliveryRes	ChargingStatusRes	MeteringReceiptRes	CertificateupdateRes	CertificateinstallationRes	SessionStopRes
OK	x	x	x	x	x	x	x	x	x	x	x	x	x	x
OK_CertificateExpiresSoon					x									
OK_NewSessionEstablished		x												
OK_OldSessionJoined		x												
FAILED	x	x	x	x	x	x	x	x	x	x	x	x	x	x
FAILED_SequenceError	x	x	x	x	x	x	x	x	x	x	x	x	x	x
FAILED_SignatureError	x	x	x	x	x	x	x	x	x	x	x	x	x	x
FAILED_UnknownSession			x	x	x	x	x	x	x	x	x	x	x	x
FAILED_ServiceIDInvalid				x										
FAILED_Payment SelectionInvalid					x									
FAILED_ServiceSelection Invalid					x									
FAILED_CertificateExpired						x						x	x	
FAILED_CertificateRevoked						x						x	x	
FAILED_NoCertificate Available						x						x	x	
FAILED_CertChainError												x		
FAILED_ContractCanceled												x		
FAILED_ChallengeInvalid							x							
FAILED_WrongEnergy TransferMode								x						
FAILED_WrongCharge Parameter								x						
FAILED_ChargingProfile Invalid									x					
FAILED_TariffSelection Invalid									x					
FAILED_PowerDelivery NotApplied									x					
FAILED_MeteringSignature NotValid											x			
FAILED_NoChargeServiceSelected					x									
FAILED_ContactorError									x					
FAILED_CertificateNotAllowedAtThisEVSE						x								

8.8.4 Требования к последовательности сообщений запрос-ответ

8.8.4.1 Общие требования

[V2G2-672] Во время сессии связи V2G SECC должен применять уникальный SAScheduleTupleIDs в параметре SAScheduleTuple.

Примечание 1 — Уникальные идентификаторы требуются для того, чтобы EVCC и SECC могли опираться на конкретный график SASchedule во время всей сессии связи V2G. Это также гарантирует, что EVCC может выбрать правильный профиль зарядки при повторном согласовании. В общем случае во время пересогласования EVCC может опять выбрать применяемый в данное время профиль зарядки или выбрать новый профиль зарядки, основанный на последнем SASchedule, представленном SECC.

[V2G2-673] EVCC должен рассчитать профиль зарядки, не нарушающий предельные значения параметров конкретного SAScheduleTuple (представленного посредством определенного TupleID), включенного в SASchedules, который был получен в последнем сообщении ChargeParameterDiscoveryRes.

[V2G2-674] Если EVCC отправляет действительный параметр ChargingProfile в сообщении PowerDeliveryReq, то EVSE должно обеспечить, чтобы профиль зарядки был выполнен при любых входных данных ChargingProfileEntryStart и ChargingProfileEntryMaxPower.

[V2G2-675] В случае пересогласования EVCC должен направить применяемый в настоящее время параметр ChargingProfile (тот же SAScheduleTupleID, что и перед повторным согласованием) либо направить новый параметр ChargingProfile SAScheduleTuple, который не нарушает предельные значения параметров конкретного SAScheduleTuple (представленного посредством определенного TupleID), включенного в SASchedules, который был получен в последнем сообщении ChargeParameterDiscoveryRes.

[V2G2-676] Если EVCC отправляет параметр ChargingProfile в сообщении PowerDeliveryReq, который либо выполняет ограничения SAScheduleTuple, представленные в параметре SASchedules, направленном SECC в последнем ChargeParameterDiscoveryRes, либо применяемый в настоящее время профиль зарядки (тот же SAScheduleTupleID, что и перед повторным согласованием), то EVSE должно обеспечить, чтобы профиль зарядки был выполнен при любых входных данных в параметре ChargingProfile.

[V2G2-677] Если SECC не направляет EVCC запрос-уведомление (Notification request), то он должен установить параметр EVSENotification в EVSEStatus в значение «None».

[V2G2-678] Если параметр EVSENotification в EVSEStatus равен None, то EVCC должен игнорировать значение параметра NotificationMaxDelay в EVSEStatus.

[V2G2-679] Если параметр EVSENotification в EVSEStatus равен StopCharging, то EV должно остановить зарядку в течение нескольких секунд задержки, значение которой указано в NotificationMaxDelay.

Примечание 2 — После индикации StopCharging SECC может остановить зарядку со стороны EVSE, например, сбросив сигнал управления или разомкнув питающие повода контактора по истечении времени задержки NotificationMaxDelay.

[V2G2-680] Если параметр EVSENotification в EVSEStatus равен ReNegotiation, то EV должно инициировать пересогласование в течение нескольких секунд задержки, представленной в NotificationMaxDelay (см. **[V2G2-521]**, **[V2G2-522]** и **[V2G2-686]**).

Примечание 3 — Решение EV относительно принятия или отказа от повторного пересогласования, инициируемого EVSE, находится за пределами данной спецификации, но зависит от конкретных сопутствующих факторов процесса зарядки, например заключенного контракта на зарядку. Пересогласование может противоречить ожиданиям пользователя, если это не согласуется с ним заранее во время первого согласования. Таким образом, пересогласование предоставляет только технические средства для поддержки случаев использования, которые должны быть заранее согласованы с пользователем.

8.8.4.2 EVCC

Поведение EVCC, определяющее все действительные последовательности сообщений запрос-ответ для переменного тока, показано на рисунке 101, а для постоянного тока — на рисунке 102.

8.8.4.2.1 Общие требования

[V2G2-482] EVCC должен прекращать сеанс связи V2G каждый раз при получении сообщения-ответа, которое не соответствует последнему отправленному сообщению-запросу.

Примечание 1 — Это означает, например, что EVCC может принимать SessionSetupRes, только если сообщение, посланное перед этим, было сообщением SessionSetupReq.

[V2G2-483] EVCC должен отправлять supportedAppProtocolReq.

[V2G2-484] EVCC должен прекращать сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout сообщения «supportedAppProtocolRes» в соответствии с таблицей 109.

[V2G2-485] После получения supportedAppProtocolRes EVCC должен отправлять SessionSetupReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-486] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «SessionSetupRes» в соответствии с таблицей 109.

[V2G2-487] После получения SessionSetupRes EVCC должен отправить ServiceDiscoveryReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-488] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «ServiceDiscoveryRes» в соответствии с таблицей 109.

[V2G2-489] После получения ServiceDiscoveryRes EVCC должен отправить ServiceDetailReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если SECC предлагает ServiceList в ServiceDiscoveryRes и EVCC намерен использовать услугу из ServiceList.

[V2G2-490] После получения ServiceDiscoveryRes EVCC должен отправить PaymentServiceSelectionReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если для дальнейшего процесса не требуется дополнительных деталей касательно сервисного обслуживания.

[V2G2-491] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «ServiceDetailRes» в соответствии с таблицей 109.

[V2G2-492] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «PaymentServiceSelectionRes» в соответствии с таблицей 109.

[V2G2-509] После получения PaymentServiceSelectionRes EVCC должен отправить AuthorizationReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-493] После получения ServiceDetailRes EVCC должен отправить PaymentServiceSelectionReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если больше не предполагается запросов ServiceDetailReq.

[V2G2-494] После получения ServiceDetailRes EVCC должен отправить дополнительный ServiceDetailReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если дальнейшие запросы Service DetailReq необходимы для получения подробной информации от SECC.

[V2G2-495] После получения PaymentServiceSelectionRes EVCC должен отправить PaymentDetailsReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если выбран набор сообщений «AC Charing PnC» или «DC Charging PnC» и EVCC не намерен использовать наборы сообщений «Certificate Install» или «Certificate Update».

[V2G2-496] После получения PaymentServiceSelectionRes EVCC должен отправить CertifiacteInstallationReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если ServiceDetailRes показывает, что установка сертификата доступна и EVCC хочет использовать набор сообщений «Certificate Installation».

[V2G2-497] После получения PaymentServiceSelectionRes EVCC должен отправить CertifiacteUpdateReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если ServiceDetailRes показывает, что обновление сертификата доступно и EVCC хочет использовать набор сообщений «Certificate Update».

[V2G2-498] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «CertifiacteInstallationRes» в соответствии с таблицей 109.

[V2G2-499] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout в сообщении «CertifiacteUpdateRes» в соответствии с таблицей 109.

[V2G2-500] После получения CertificateInstallationRes EVCC должен отправить PaymentDetailsReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-501] После получения CertificateUpdateRes EVCC должен отправить PaymentDetailsReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-502] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «PaymentDetailsRes» в соответствии с таблицей 109.

[V2G2-503] После получения PaymentDetailsRes EVCC должен отправить AuthorizationReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-504] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «AuthorizationRes» в соответствии с таблицей 109.

[V2G2-505] После получения AuthorizationRes с параметром «EVSEProcessing», установленным в значение «Finished», EVCC должен отправить ChargeParameterDiscoveryReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-506] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «ChargeParameterDiscoveryRes» в соответствии с таблицей 109.

[V2G2-799] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «PowerDeliveryRes» в соответствии с таблицей 109.

[V2G2-507] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILE» в сообщении «SessionStopRes» в соответствии с таблицей 109.

[V2G2-508] После получения SessionStopRes EVCC должен прекратить связь путем применения **[V2G2-025]**.

[V2G2-728] После того как EVCC остановил сессию связи V2G с ошибкой, он должен прекратить коммуникацию путем применения **[V2G2-025]**.

EVCC может пересогласовать график зарядки следующим образом:

[V2G2-683] Если EVCC инициировал пересогласование (см. **[V2G2-521]**, **[V2G2-522]** и **[V2G2-686]**) и получает PowerDeliveryRes, то EVCC должен отправить ChargeParameterDiscoveryReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

Если SECC приостанавливает последовательность сообщений посредством EVSEProcessing, установленным в значение «Ongoing», к EVCC применяется следующее:

[V2G2-684] После получения AuthorizationRes EVCC должен отправлять пустой AuthorizationReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time при условии, что EVSEProcessing равен «Ongoing».

Примечание 2 — EV отправляет сообщение AuthorizationReq, содержащее Signature, Id и GenChallenge в составе только первого сообщения AuthorizationReq. Последующие сообщения посылаются без этой информации, если сообщение AuthorizationRes было получено с EVSEProcessing, установленным в значение «Ongoing».

[V2G2-685] После получения ChargeParameterDiscoveryRes EVCC должен повторно посылать ChargeParameterDiscoveryReq при V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time до тех пор, пока EVSEProcessing равен «Ongoing».

8.8.4.2.2 Специфичные требования для переменного тока

[V2G2-510] После получения ChargeParameterDiscoveryRes с параметром «EVSEProcessing», установленным в значение «Finished», EVCC должен отправить PowerDeliveryReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-511] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «ChargingStatusRes» в соответствии с таблицей 109.

[V2G2-512] После получения ChargingStatusRes EVCC должен отправить MeteringReceiptReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если выбран набор сообщений «Meter Status Receipt», указываемый, если параметр «Receipt Required» в сообщении «ChargingStatusRes» устанавливается в значение «TRUE».

[V2G2-513] После получения ChargingStatusRes EVCC должен отправить PowerDeliveryReq с параметром ChargProgress, установленным в значение «Stop», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки в EV остановлен и набор сообщений «Meter Status Receipt» не выбран. Это означает, что параметр «Receipt Required» в сообщении «ChargingStatusRes» был установлен в значение «FALSE».

[V2G2-514] После получения PowerDeliveryRes EVCC должен отправить ChargingStatusReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-516] После получения ChargingStatusRes EVCC должен отправить ChargingStatusReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки продолжается и параметр «ReceiptRequired» в сообщении «ChargingStatusRes» был установлен в значение «FALSE».

[V2G2-517] EVCC должен прекратить сеанс связи V2G, если V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «MeteringReceiptRes» в соответствии с таблицей 109.

[V2G2-518] После получения MeteringReceiptRes EVCC должен отправить ChargingStatusReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки продолжается.

[V2G2-519] После получения MeteringReceiptRes EVCC должен отправить PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки в EV остановлен.

[V2G2-520] После получения PowerDeliveryRes EVCC должен отправить SessionStopReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если был отослан PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop».

EVCC может пересогласовать график зарядки следующим образом:

[V2G2-521] EV может произвести повторное согласование для адаптации своего графика зарядки следующим образом. После того как ChargingStatusRes был получен и параметр «Receipt Required» установлен в значение «FALSE», EVCC должен инициировать пересогласование посредством отсылки PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiate», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-522] EV может произвести повторное согласование для адаптации своего графика зарядки следующим образом. После того как MeteringReceiptRes получен, EVCC должен инициировать пересогласование посредством отсылки PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiate», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-689] Если параметр EVSENotification в EVSEStatus равен ReNegotiation в ChargingStatusRes, то EV должен инициировать пересогласование, как описано в **[V2G2-521]** и **[V2G2-522]**, в течение интервала времени в несколько секунд, определенного в NotificationMaxDelay.

Примечание 1 — EV может произвести повторное согласование, применяя процесс, описанный в **[V2G2-521]** и **[V2G2-522]**. Пересогласование, запускаемое EVSE, в своей основе — это тот же самый процесс, что и процесс пересогласования, инициируемый EV, но процесс, запускаемый EVSE, описывается в **[V2G2-689]**.

Примечание 2 — Решение инициировать пересогласование со стороны EV принимается самим EV. Например, если график зарядки ChargingProfile, уже одобренный EVCC, должен быть изменен, исходя из требований удобства пользователя или к оборудованию сети, то EVCC может запустить процесс пересогласования, подразумевающий повторный обмен SASchedule и ChargingProfile и достижение соглашения с SECC.

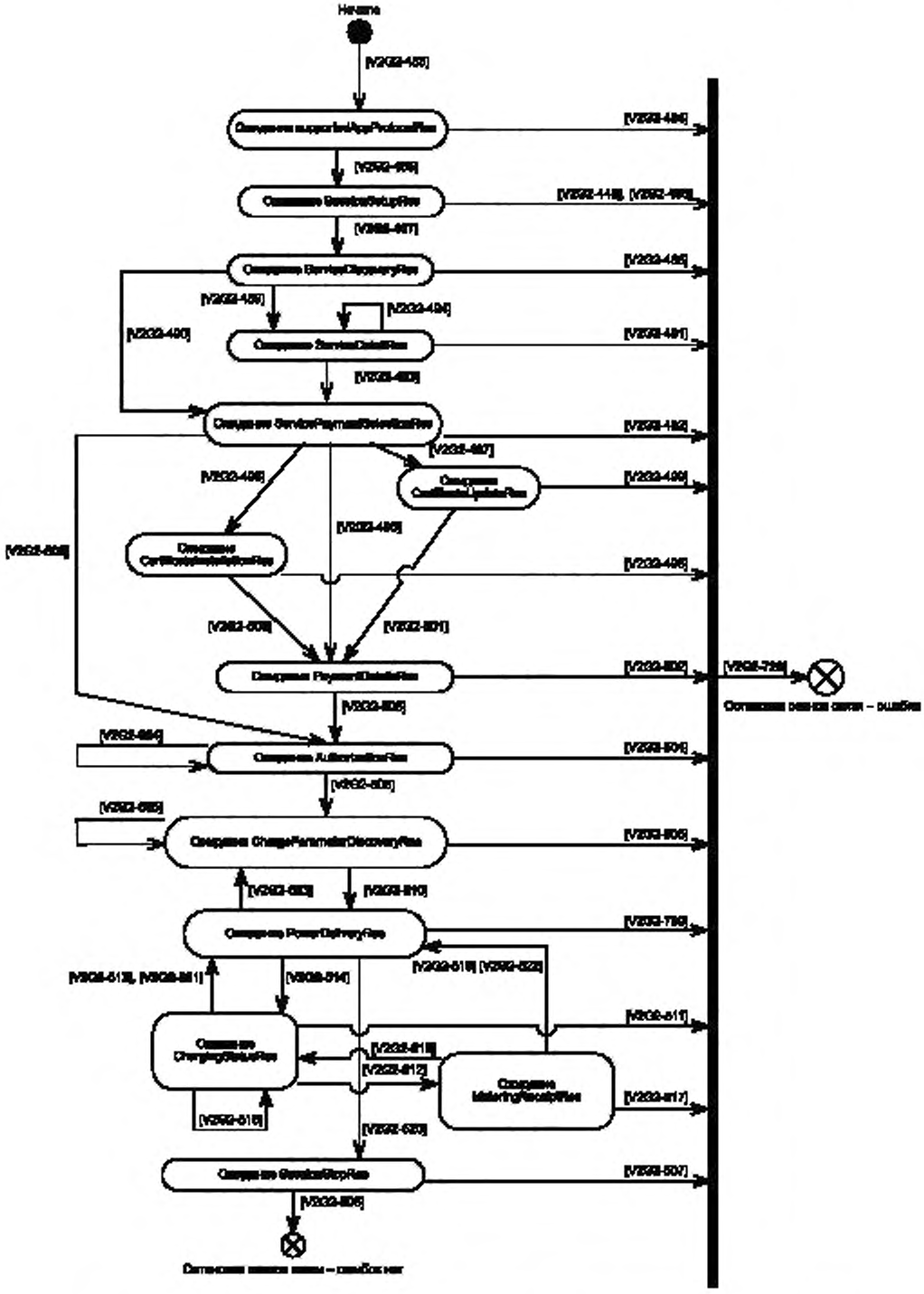


Рисунок 101 — Состояния связи EVCC для сообщений V2G при зарядке переменным током

8.8.4.2.3 Специфичные требования для постоянного тока

[V2G2-880] Все коды DC_EVSEStatusCodes, явно заданные требования к которым в настоящем стандарте отсутствуют, приведены в таблице 98 только в качестве справочной информации. Они не должны влиять на процесс зарядки, обеспечиваемый EVSE.

Примечание 1 — «EVSE_EmergencyShutdown» может использоваться SECC для сообщения EVCC о том, что EVSE находится в процессе или только что выполнило процесс аварийного отключения. Однако для того, чтобы заставить EV принять участие в аварийном отключении, SECC не использует это значение, а взамен использует линию управления (CPL) в соответствии с *ГОСТ Р МЭК 61851-1*. Во всех других случаях ожидаемым значением DC_EVSEStatusCode является «EVSE_Ready» или «EVSE_IsolationMonitoringActive», даже несмотря на то что они не влияют на процесс зарядки EV.

[V2G2-599] После получения ChargeParameterDiscoveryRes EVCC должен отправить CableCheckReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-524] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILE» в сообщении «CableCeckRes» в соответствии с таблицей 109.

[V2G2-617] После получения CableCheckRes EVCC должен отправить новое сообщение CableCheckReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time при условии, что параметр EVSEProcessing равен «Ongoing».

[V2G2-525] После получения CableCeckRes с параметром «EVSEProcessing», установленным в значение «Finished», EVCC должен отправить PreChargeReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-526] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «PreChargeReq» в соответствии с таблицей 109.

[V2G2-527] После получения CurrentDemandRes EVCC должен отправить PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки остановлен.

[V2G2-618] После получения PreChargeRes EVCC должен отправить PreChargeReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, и значение параметра EVSEPresentVoltage не соответствует требованию EV по порогу напряжения.

Примечание 2 — EV может использовать внутренние методы измерения напряжения для оценки входного напряжения, получаемого в сообщении PreChargeRes (EVSEPresentVoltage).

[V2G2-528] После получения PreChargeRes EVCC должен отправить PowerDeliveryReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time и значение параметра EVSEPresentVoltage соответствует требованию EV по порогу напряжения.

[V2G2-530] После получения PowerDeliveryRes EVCC должен отправить CurrentDemandReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

[V2G2-619] После получения PowerDeliveryRes как ответа на предшествующее сообщение PowerDeliveryReq с параметром ChargeProgress, равным «Stop», EVCC должен отправить SessionStopReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если EV не определяет сваривание контактов.

[V2G2-531] После получения CurrentDemandRes с параметром «ReceiptRequired», установленным в значение «FALSE», EVCC должен отправить CurrentDemandReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если EV продолжает процесс зарядки.

[V2G2-790] После получения CurrentDemandRes EVCC должен отправить MeteringReceiptReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_

Time, если параметр «ReceiptRequired» в сообщении CurrentDemandRes установлен в значение «TRUE».

[V2G2-791] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAIL» в сообщении «MeteringReceiptRes» в соответствии с таблицей 109.

[V2G2-792] После получения MeteringReceiptRes EVCC должен отправить CurrentDemandReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если процесс зарядки продолжается.

[V2G2-793] После получения MeteringReceiptRes EVCC должен отправить PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Stop», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если EV прекращает процесс зарядки.

[V2G2-532] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «CurrentDemandRes» в соответствии с таблицей 109.

[V2G2-533] После получения PowerDeliveryRes в ответ на предшествующее сообщение PowerDeliveryReq с ChargeProgress, равным «Stop», EVCC должен отправить WeldingDetectionReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, если EV определяет сваривание контактов.

[V2G2-534] EVCC должен прекратить сеанс связи V2G, когда V2G_EVCC_Msg_Timer больше или равен V2G_EVCC_Msg_Timeout либо «ResponseCode = FAILED» в сообщении «WeldingDetectionRes» в соответствии с таблицей 109.

[V2G2-620] После получения WeldingDetectionRes EVCC должен отправить WeldingDetectionReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time, и выполнение функции определения сваривания контактов не завершено со стороны EV.

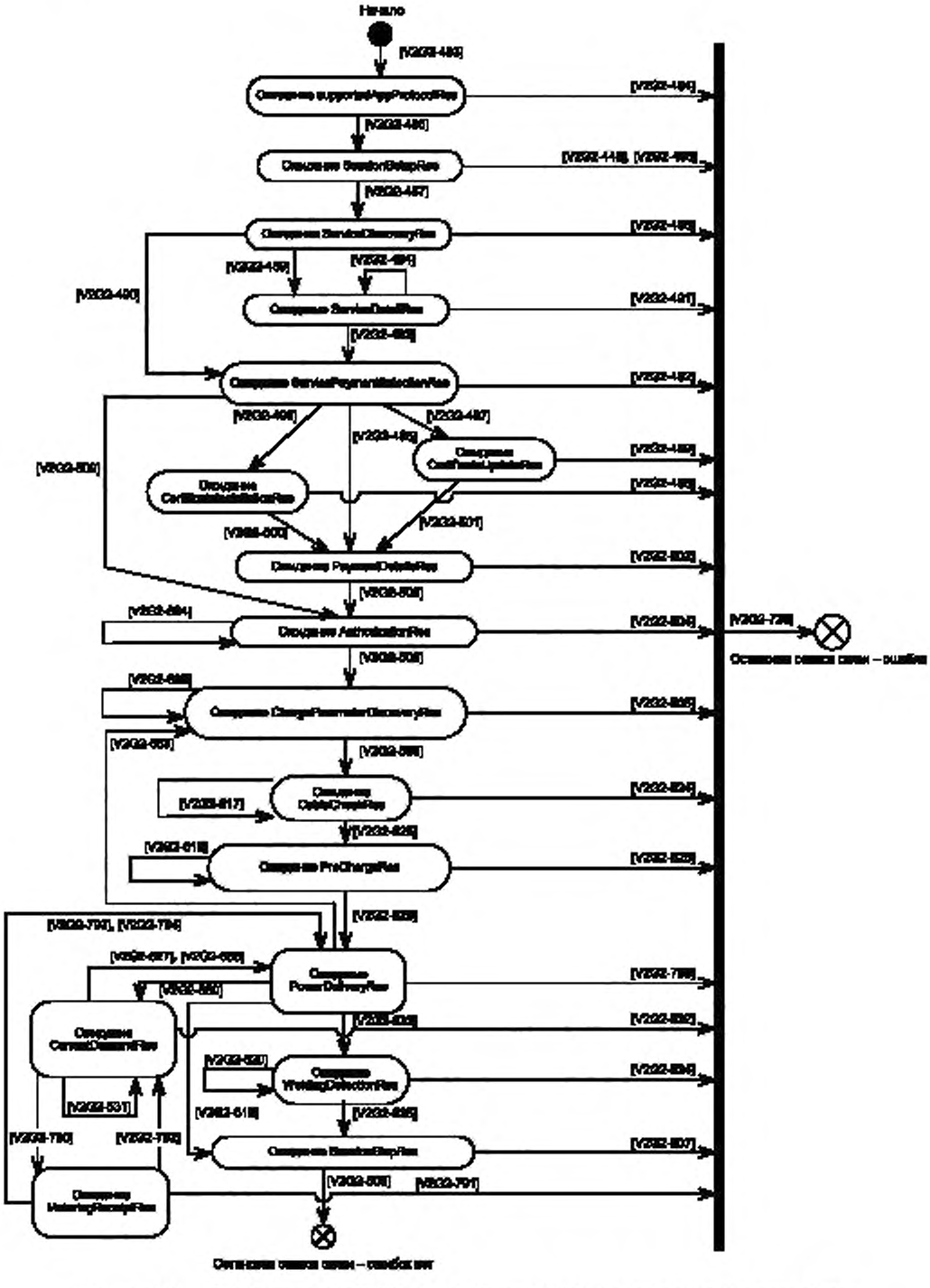
[V2G2-535] После получения WeldingDetectionRes EVCC должен отправить SessionStopReq, пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time.

EVCC может повторно согласовать график зарядки следующим образом:

[V2G2-686] EV может принять решение произвести пересогласование для адаптации своего графика зарядки следующим образом: после получения ответа CurrentDemandRes и согласования EVCC графика зарядки последний должен инициировать пересогласование посредством отсылки PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiate», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time*.

[V2G2-794] EV может принять решение о выполнении пересогласования для адаптации своего графика зарядки следующим образом: после получения ответа CurrentDemandRes и согласования EVCC графика зарядки последний должен инициировать пересогласование посредством отсылки PowerDeliveryReq с параметром ChargeProgress, установленным в значение «Renegotiate», пока V2G_EVCC_Sequence_Timer меньше, чем V2G_EVCC_Sequence_Performance_Time*.

* [V2G2-686] и [V2G2-794] идентичны, как в оригинале ИСО 15118-2:2013.


Начало
[V2G2-483]
[V2G2-484]
[V2G2-485]
[V2G2-486]
[V2G2-487]
[V2G2-488]
[V2G2-489]
[V2G2-490]
[V2G2-491]
[V2G2-492]
[V2G2-493]
[V2G2-494]
[V2G2-495]
[V2G2-496]
[V2G2-497]
[V2G2-498]
[V2G2-499]
[V2G2-500]
[V2G2-501]
[V2G2-502]
[V2G2-503]
[V2G2-504]
[V2G2-505]
[V2G2-506]
[V2G2-507]
[V2G2-508]
[V2G2-509]
[V2G2-510]
[V2G2-511]
[V2G2-512]
[V2G2-513]
[V2G2-514]
[V2G2-515]
[V2G2-516]
[V2G2-517]
[V2G2-518]
[V2G2-519]
[V2G2-520]
[V2G2-521]
[V2G2-522]

8.8.4.3 SECC

Режим SECC, определяющий все действительные последовательности сообщений запрос-ответ для переменного тока, показан на рисунке 103, а для постоянного тока — на рисунке 104.

8.8.4.3.1 Общие требования

[V2G2-536] SECC должен войти в состояние ожидания сообщения supportedAppProtocolReq, установить тайм-аут V2G_SECC_Sequence_Timeout в значение MessageType, указанное в таблице 109, сбросить V2G_SECC_Sequence_Timer и начать мониторинг V2G_SECC_Sequence_Timer.

Примечание — До первого сообщения SECC не направлял сообщения-ответа. Таким образом, SECC должен запустить свой таймер последовательности (Sequence Timer), начиная с ожидания первого сообщения.

[V2G2-537] SECC должен прекратить сеанс связи V2G, если V2G_SECC_Sequence_Timer больше или равен V2G_SECC_Sequence_Timeout в соответствии с таблицей 109.

[V2G2-538] SECC должен направить сообщение-ответ, содержащее «ResponseCode = FAILED_SequenceError», в течение времени V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если было получено сообщение-запрос, которое SECC не предполагал получить в состоянии ожидания.

[V2G2-539] После того как SECC послал «ResponseCode = FAILED», он должен прекратить связь согласно **[V2G2-034]**.

[V2G2-729] После того как SECC прекратил сеанс связи V2G, он должен прекратить связь согласно **[V2G2-034]**.

[V2G2-540] После получения запроса supportedAppProtocolReq SECC должен обработать полученную информацию.

[V2G2-541] SECC должен ответить сообщением supportedAppProtocolRes в течение времени V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109. Следующим разрешенным запросом должен быть ServiceSetupReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-542] После получения SessionSetupReq SECC должен обработать полученную информацию.

[V2G2-543] SECC должен ответить сообщением SessionSetupRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109. Следующим разрешенным запросом должен быть ServiceDiscoveryReq, и V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-544] После получения ServiceDiscoveryReq SECC должен обработать полученную информацию.

[V2G2-545] SECC должен ответить сообщением ServiceDiscoveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть ServiceDetailReq и PaymentServiceSelectionReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-546] SECC должен ответить сообщением ServiceDiscoveryRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-547] После получения ServiceDetailReq SECC должен обработать полученную информацию.

[V2G2-548] SECC должен ответить сообщением ServiceDetailRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть ServiceDetailReq и PaymentServiceSelectionReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-549] SECC должен ответить сообщением ServiceDetailRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-550] После получения PaymentServiceSelectionReq SECC должен обработать полученную информацию.

[V2G2-551] SECC должен ответить сообщением PaymentServiceSelectionRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть PaymentDetailsReq, CertificateInstallationReq и CertificateUpdateReq, если выбран набор сообщений «AC Charging PnC», и AuthorizationReq, если выбран набор сообщений «AC Charging EIM». V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-552] SECC должен ответить сообщением PaymentServiceSelectionRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-553] После получения CertificateInstallationReq SECC должен обработать полученную информацию.

[V2G2-554] SECC должен ответить сообщением CertificateInstallationRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующим разрешенным запросом должен быть PaymentDetailsReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-555] SECC должен ответить сообщением CertificateInstallationRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-556] После получения CertificateUpdateReq SECC должен обработать полученную информацию.

[V2G2-557] SECC должен ответить сообщением CertificateUpdateRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующим разрешенным запросом должен быть PaymentDetailsReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-558] SECC должен ответить сообщением CertificateUpdateRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не прошла успешно. Следующим разрешенным запросом должен быть PaymentDetailsReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-559] После получения PaymentDetailsReq SECC должен обработать полученную информацию.

[V2G2-560] SECC должен ответить сообщением PaymentDetailsRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующим разрешенным запросом должен быть AuthorizationReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-561] SECC должен ответить сообщением PaymentDetailsRes, содержащим ResponseCode = FAILED», в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109.

[V2G2-562] После получения AuthorizationReq SECC должен обработать полученную информацию.

[V2G2-563] SECC должен ответить сообщением AuthorizationRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Finished», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и авторизация завершена. Следующим разрешенным запросом должен быть ChargeParameterDiscoveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-564] SECC должен ответить сообщением AuthorizationRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не прошла успешно.

[V2G2-566] SECC должен ответить сообщением ChargeParameterDiscoveryRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не прошла успешно.

[V2G2-567] После получения PowerDeliveryReq SECC должен обработать полученную информацию.

[V2G2-568] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и запрос содержал ChargeProgress, установленный в значение «Stop». Следующим разрешенным запросом должен быть SessionStopReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-569] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-812] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно, запрос содержал ChargeProgress, установленный в значение «Renegotiate», и перед этим не был получен запрос PowerDeliveryReq с ChargeProgress, установленным в значение «Start».

[V2G2-570] После получения SessionStopReq SECC должен обработать полученную информацию и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-571] SECC должен ответить сообщением SessionStopRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109 и должен прекратить связь путем применения **[V2G2-034]**, если обработка информации прошла успешно. После этого сеанс связи прекращается без фиксирования ошибки.

[V2G2-572] SECC должен ответить сообщением SessionStopRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-687] SECC должен ответить сообщением AuthorizationRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Ongoing», в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и авторизация продолжается. Следующим разрешенным запросом должен быть AuthorizationReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-688] SECC должен ответить сообщением ChargeParameterDiscoveryRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Ongoing» без параметра SASchedule, в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно, а вычисление параметра SASchedule продолжается. Следующим разрешенным запросом должен быть ChargeParameterDiscoveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

EVCC может пересогласовать график зарядки следующим образом:

[V2G2-813] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и запрос содержал ChargeProgress, установленный в значение «Renegotiate». Следующим разрешенным запросом должен быть ChargeParameterDiscoveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

8.8.4.3.2 Специфичные требования для переменного тока

[V2G2-573] SECC должен ответить сообщением ChargeParameterDiscoveryRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Finished» и действительный параметр SASchedule, в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующим разрешенным

запросом должен быть PowerDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-574] После получения ChargingStatusReq SECC должен обработать полученную информацию.

[V2G2-575] SECC должен ответить сообщением ChargingStatusRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. SECC устанавливает «ReceiptRequired = FALSE», показывая, что набор сообщений MessageReceipt не должен использоваться EVCC. Следующими разрешенными запросами должны быть ChargingStatusReq, PowerDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-576] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и запрос содержал ChargeProgress, установленный в значение «Start». Следующим разрешенным запросом должен быть ChargingStatusReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-577] SECC должен ответить сообщением ChargingStatusRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. SECC устанавливает «ReceiptRequired = TRUE», показывая, что набор сообщений MessageReceipt должен использоваться EVCC. Следующим разрешенным запросом должен быть MeteringReceiptReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-578] SECC должен ответить сообщением ChargingStatusRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-579] После получения MeteringReceiptReq SECC должен обработать полученную информацию.

[V2G2-580] SECC должен ответить сообщением MeteringReceiptRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть ChargingStatusReq, PowerDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-581] SECC должен ответить сообщением MeteringReceiptRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не выполнена.

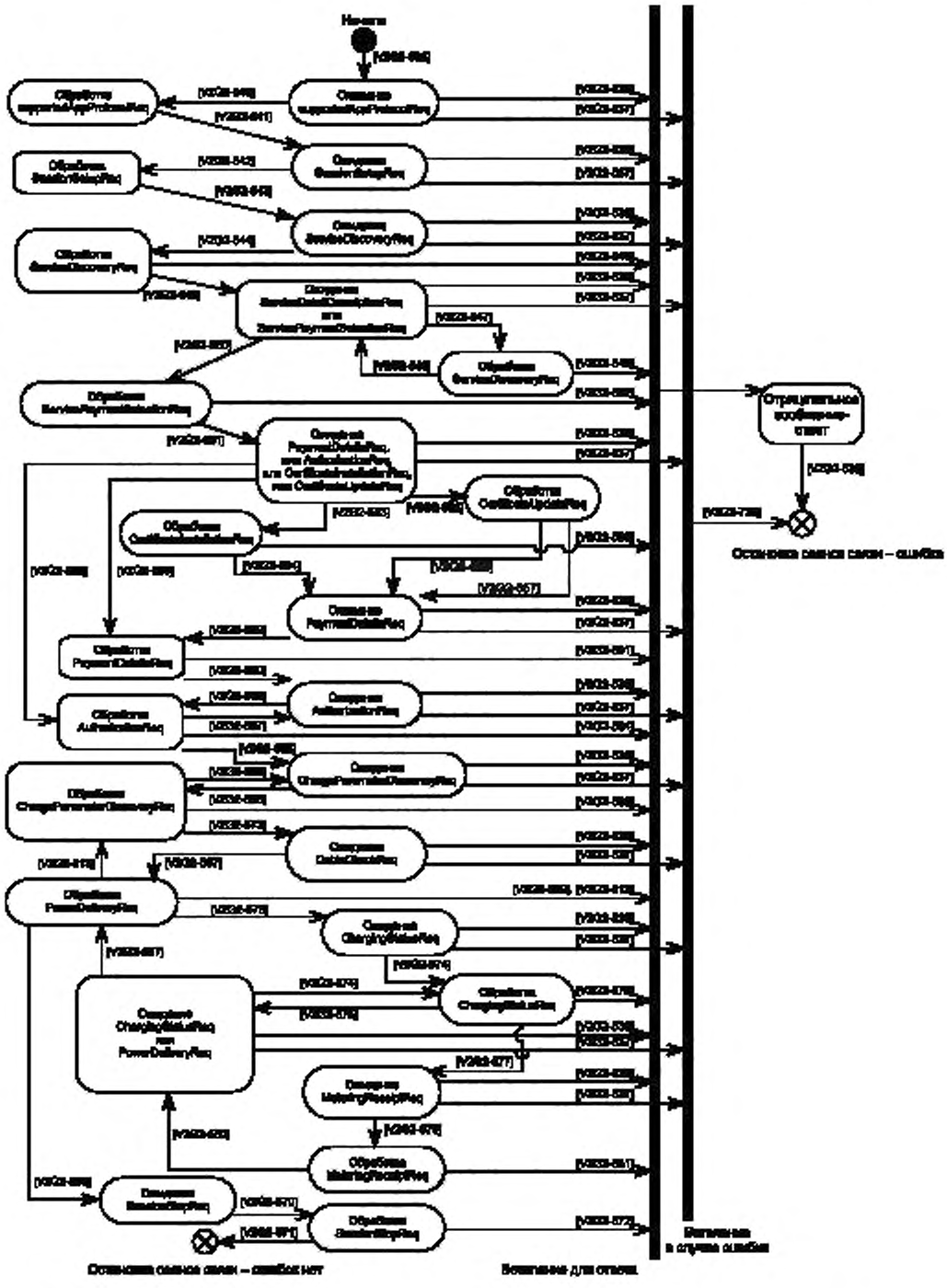


Рисунок 103 — Состояния связи SECC для сообщений V2G в случае зарядки переменным током

8.8.4.3.3 Специфичные требования для постоянного тока

[V2G2-881] Все коды ошибок (EVErrorCodes) в таблице 104, для которых в настоящем стандарте отсутствуют явно выраженные требования, приведены только в качестве справочной информации. Они могут использоваться для информации потребителя, но не должны влиять на процесс зарядки EVSE.

Примечание — SECC не должен изменять алгоритм функционирования на основании значения кода ошибки (EVErrorCode). Если EV фиксирует условие, требующее прекращения процесса зарядки, оно должно использовать другие средства для отключения, например запрос на постепенное снижение тока, или в случае аварийного отключения переход управления в состояние B в соответствии с *ГОСТ Р МЭК 61851-1*. Во всех других случаях ожидаемым значением EVErrorCode является «NO_ERROR».

[V2G2-582] SECC должен ответить сообщением ChargeParameterDiscoveryRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Finished», а также действующий параметр SASchedule, в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующим разрешенным запросом должен быть CableCheckReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-583] После получения CableCheckReq SECC должен обработать полученную информацию.

[V2G2-584] SECC должен ответить сообщением CableCheckRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Finished», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и проверка кабеля закончена. Следующим разрешенным запросом должен быть PreChargeReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-621] SECC должен ответить сообщением CableCheckRes, содержащим «ResponseCode = OK» и «EVSEProcessing = Ongoing», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и проверка кабеля продолжается. Следующим разрешенным запросом должен быть CableCheckReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-585] SECC должен ответить сообщением CableCheckRes, содержащим «ResponseCode = FAILED», в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-586] После получения PrechargeReq SECC должен обработать полученную информацию и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-587] SECC должен ответить сообщением PreChargeRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть PrechargeReq и PowtrDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-588] SECC должен ответить сообщением PreChargeRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не была успешной.

[V2G2-589] После получения PowerDeliveryReq SECC должен обработать полученную информацию и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-590] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и параметр запроса ChargeProgress установлен в значение «Start». Следующими разрешенными запросами должны быть CurrentDemmandReq и PowerDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-601] SECC должен ответить сообщением PowerDeliveryRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно и параметр запроса ChargeProgress установлен в значение «Stop».

Следующими разрешенными запросами должны быть ChargeParameterDiscoveryReq, WeldingDetectionReq и SessionStopReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-592] После получения CurrentDemandReq SECC должен обработать полученную информацию и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-593] SECC должен ответить сообщением CurrentDemandRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. SECC устанавливает «ReceiptRequired = FALSE», показывая, что набор сообщений MessageReceipt не должен использоваться EVCC. Следующими разрешенными запросами должны быть CurrentDemandReq и PowerDeliveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-595] SECC должен ответить сообщением CurrentDemandRes, содержащим «ResponseCode = FAILED», в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не выполнена.

[V2G2-795] SECC должен ответить сообщением CurrentDemandRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. SECC устанавливает «ReceiptRequired = TRUE», показывая, что набор сообщений MessageReceipt должен использоваться EVCC. Следующим разрешенным запросом должен быть MeteringReceiptReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-796] После получения MeteringReceiptReq SECC должен обработать полученную информацию.

[V2G2-797] SECC должен ответить сообщением MeteringReceiptRes, содержащим «ResponseCode = OK», в течение V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть CurrentDemandReq, PowerDeliveryReq и ChargeParameterDiscoveryReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-798] SECC должен ответить сообщением MeteringReceiptRes, содержащим «ResponseCode = FAIL», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не выполнена.

[V2G2-596] После получения WeldingDetectionReq SECC должен обработать полученную информацию и запустить V2G_SECC_Msg_Performance_Timer.

[V2G2-597] SECC должен ответить сообщением WeldingDetectionRes, содержащим «ResponseCode = OK», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации прошла успешно. Следующими разрешенными запросами должны быть WeldingDetectionReq и SessionStopReq. V2G_SECC_Sequence_Timeout устанавливается в соответствии с таблицей 109.

[V2G2-598] SECC должен ответить сообщением WeldingDetectionyRes, содержащим «ResponseCode = FAILED», в течение времени V2G_SECC_Msg_Performance_Time и в соответствии с таблицей 109, если обработка информации не выполнена.

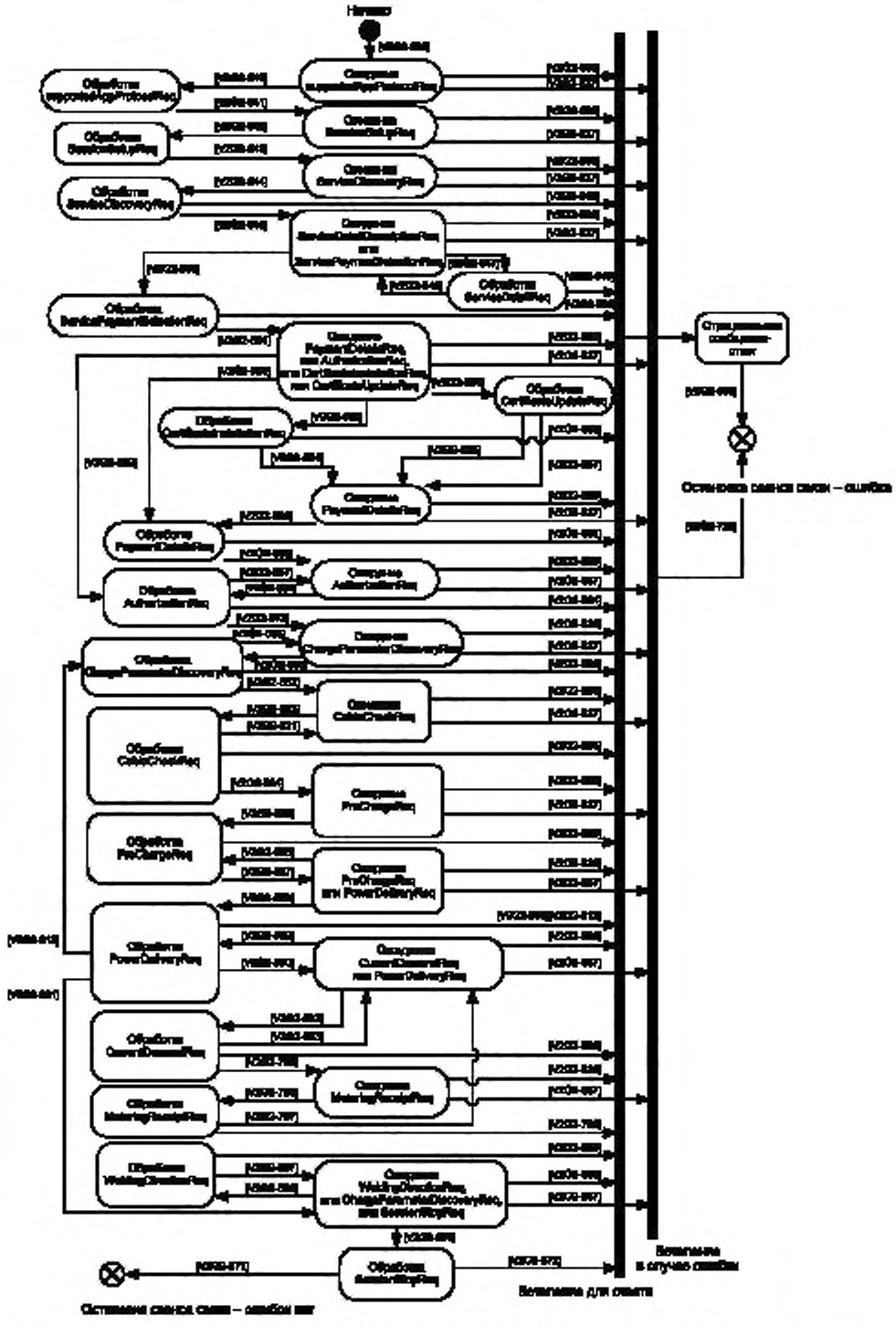


Рисунок 104 — Состояния связи SECC для сообщений V2G в случае зарядки постоянным током

8.9 Примеры последовательностей сообщений запрос-ответ

8.9.1 Переменный ток

8.9.1.1 EIM

На рисунке 105 показан пример последовательности сообщений запрос-ответ в идентификационном режиме EIM без ошибок с факультативными сообщениями.

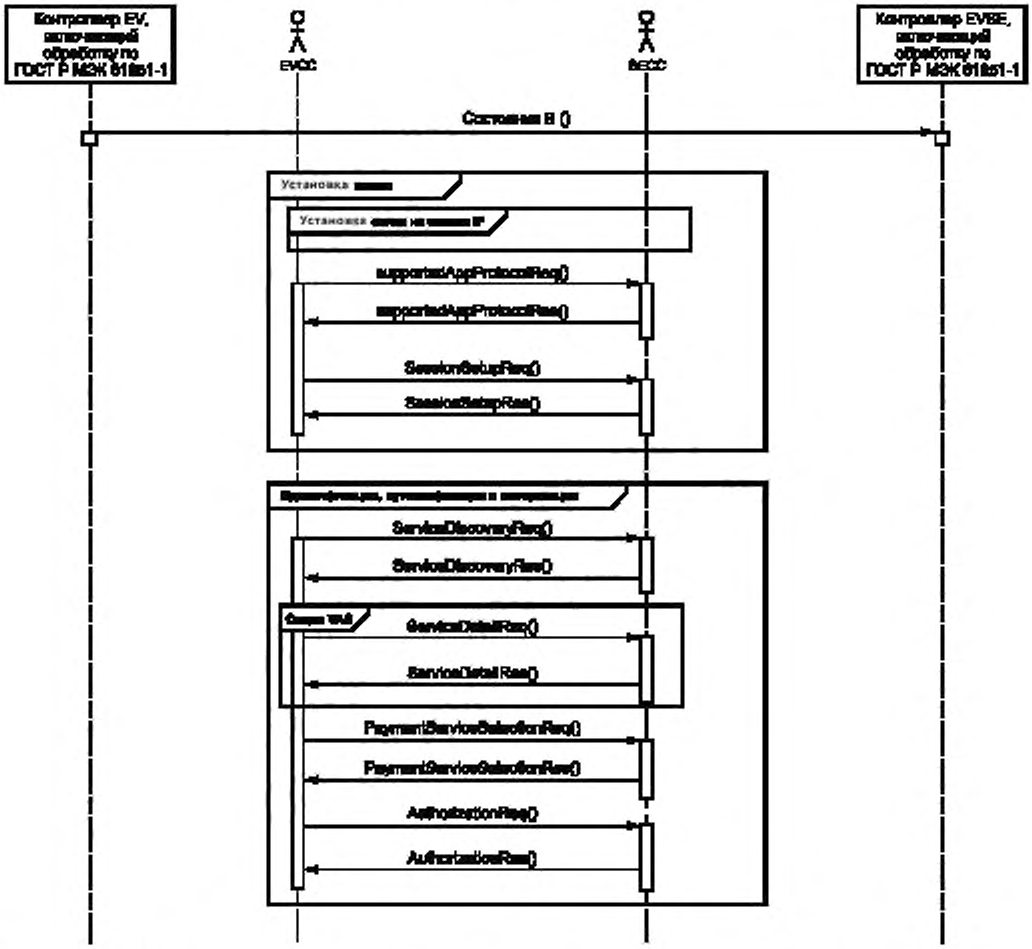


Рисунок 105 — Обзор последовательности сообщений запрос-ответ для переменного тока, идентификационный режим EIM (1 из 2)

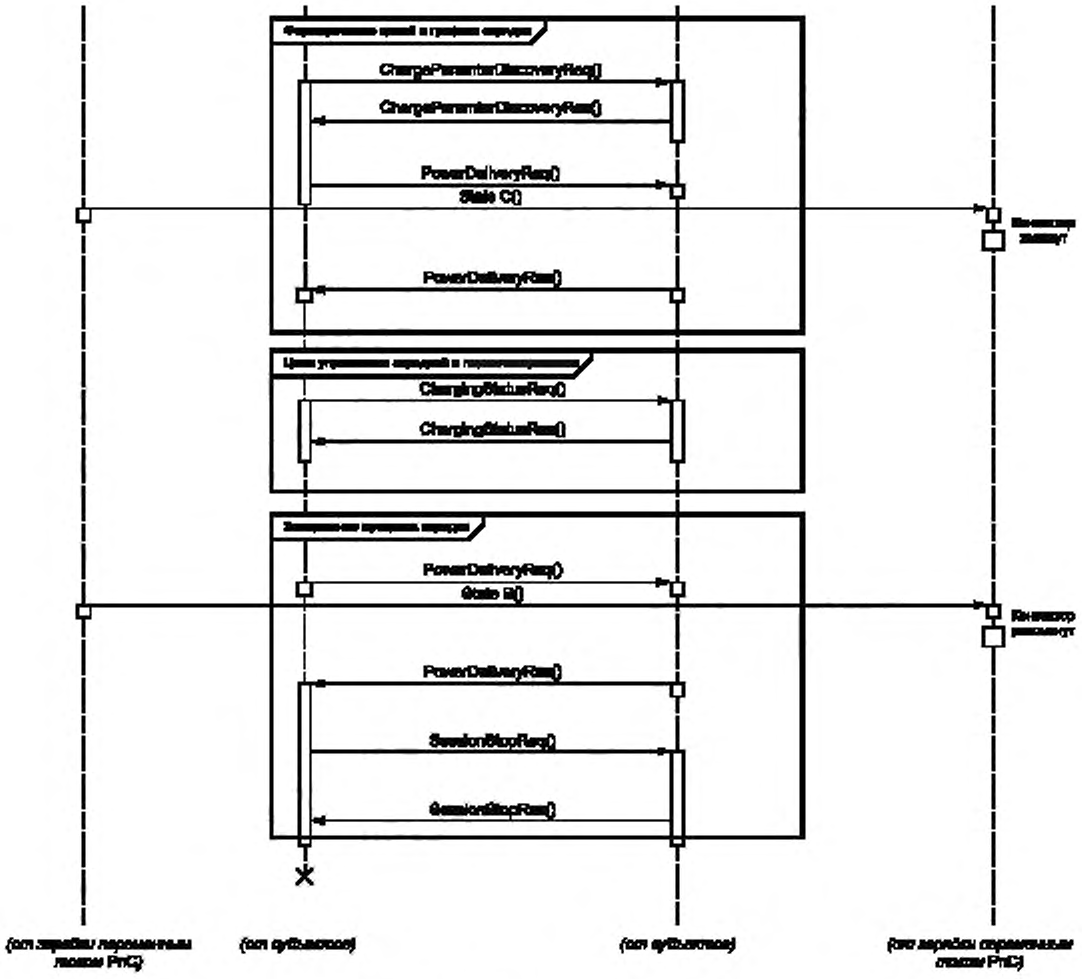


Рисунок 105, лист 2

8.9.1.2 PnC

На рисунке 106 показан пример последовательности сообщений запрос-ответ в идентификационном режиме PnC без ошибок, включая факультативные сообщения.

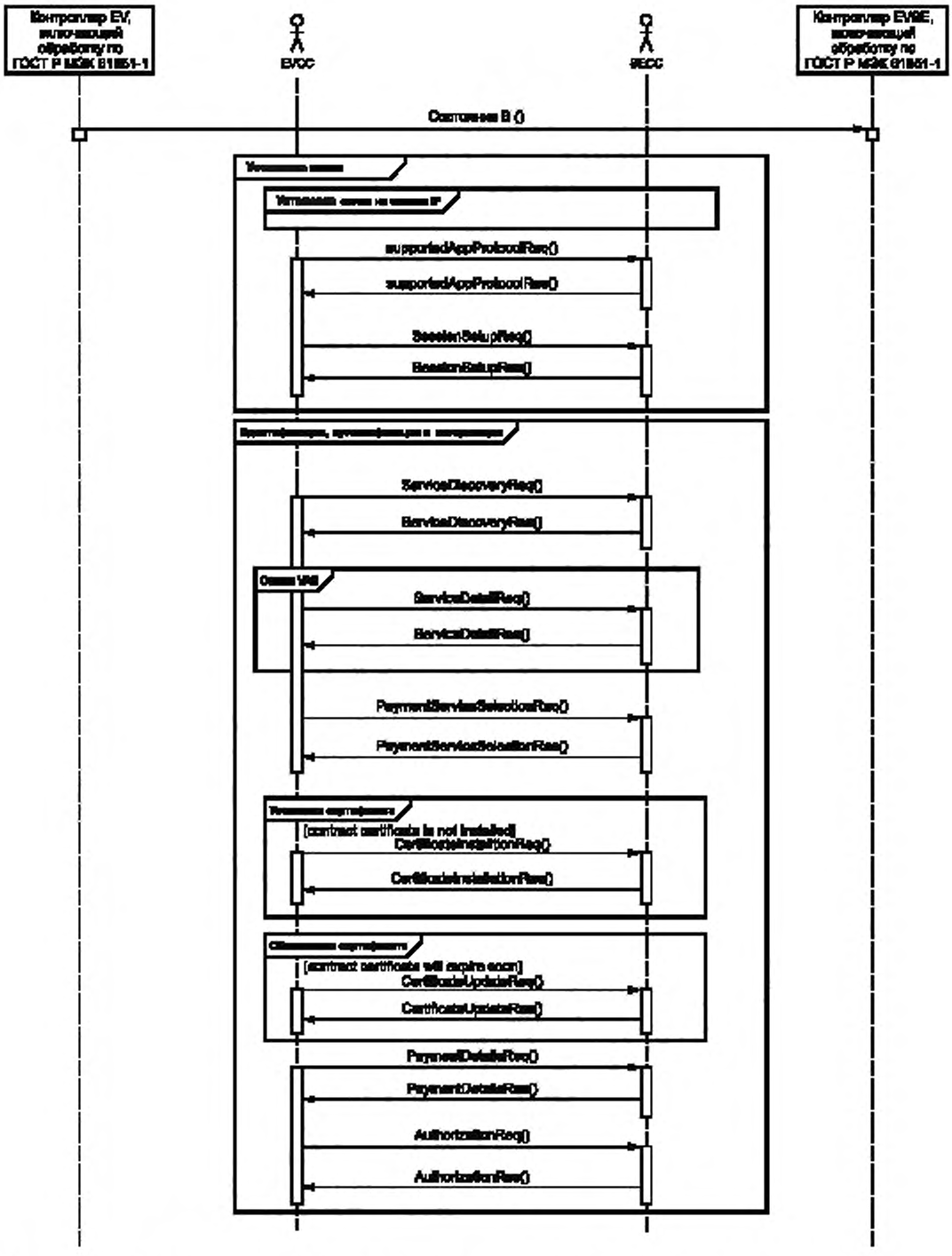


Рисунок 106 — Обзор последовательности сообщений запрос-ответ для переменного тока, идентификационный режим PnC (1 из 2)

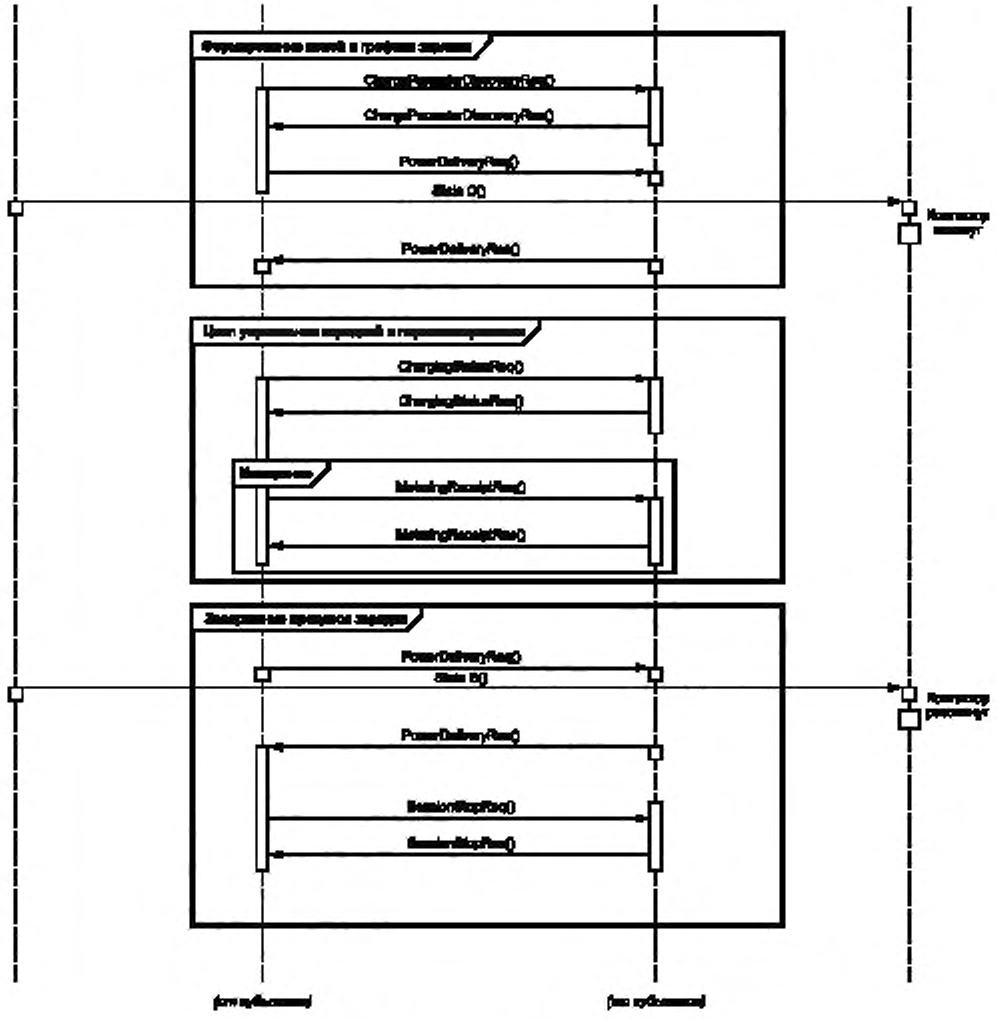


Рисунок 106, лист 2

8.9.2 Постоянный ток

8.9.2.1 EIM

На рисунке 107 показан пример последовательности сообщений запрос-ответ в идентификационном режиме EIM без ошибок, включая факультативные сообщения.

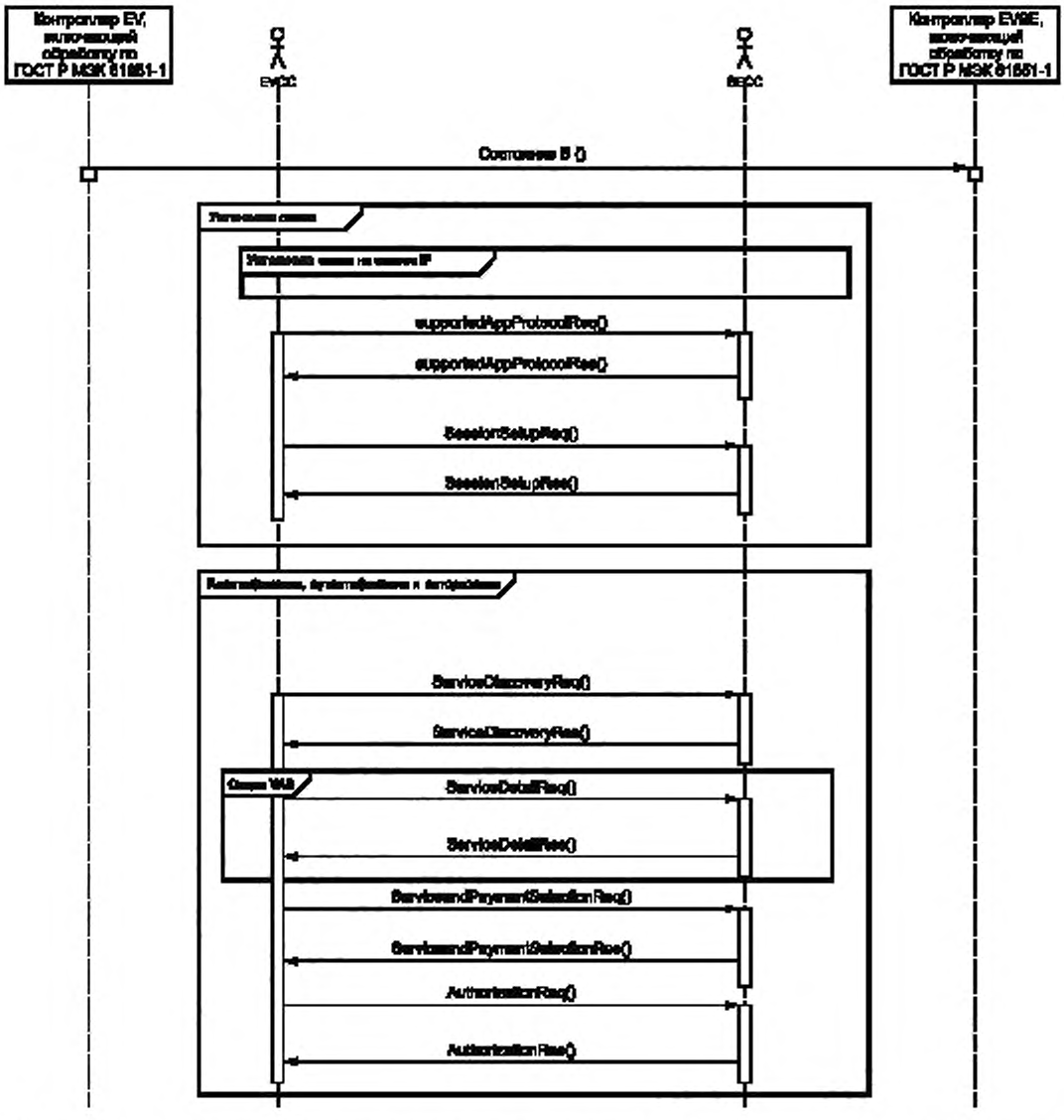


Рисунок 107 — Обзор последовательности сообщений запрос-ответ для постоянного тока, идентификационный режим EIM (1 из 2)

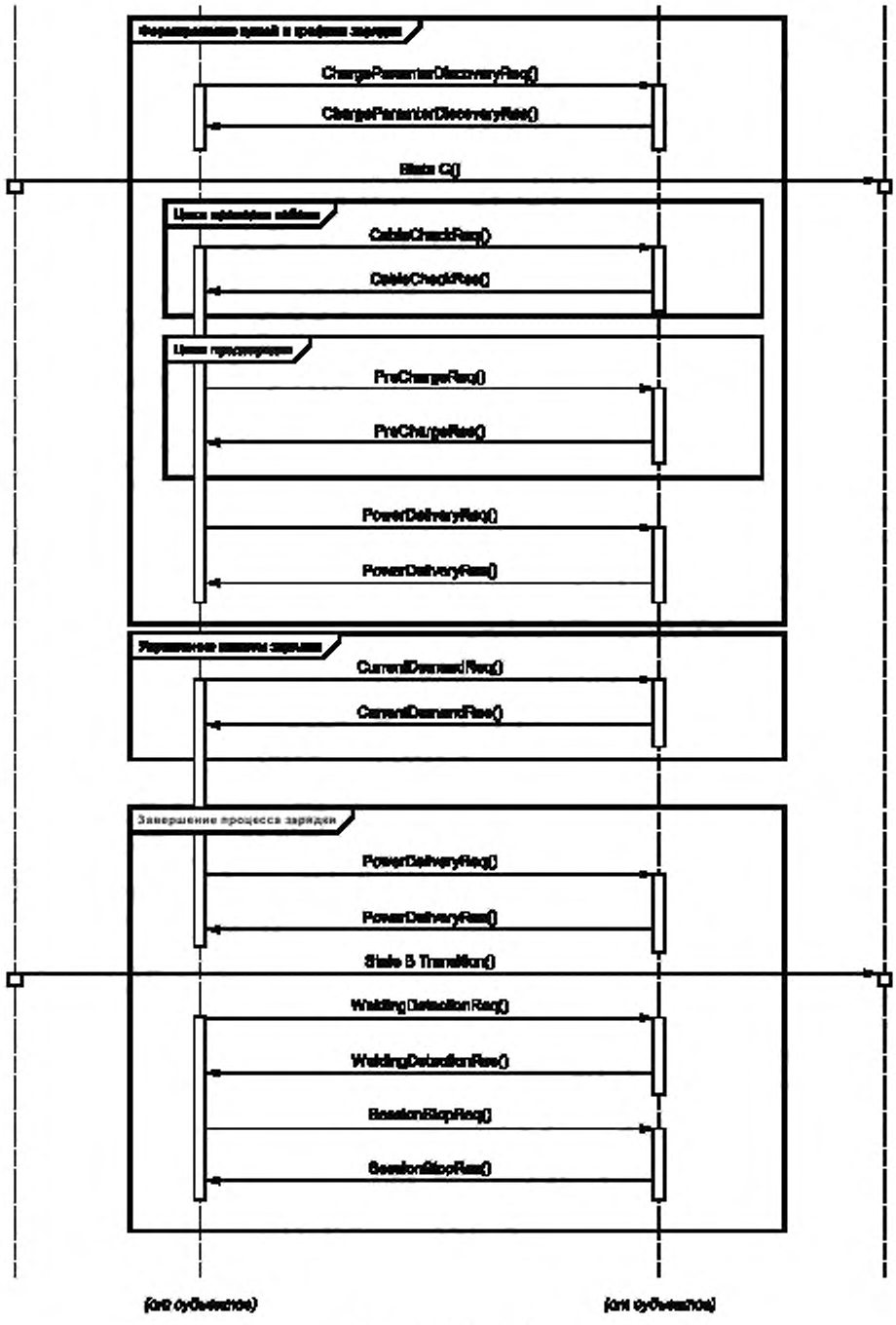


Рисунок 107, лист 2

8.9.2.2 PnC

На рисунке 108 показан пример последовательности сообщений запрос-ответ в идентификационном режиме PnC без ошибок, включая факультативные сообщения.

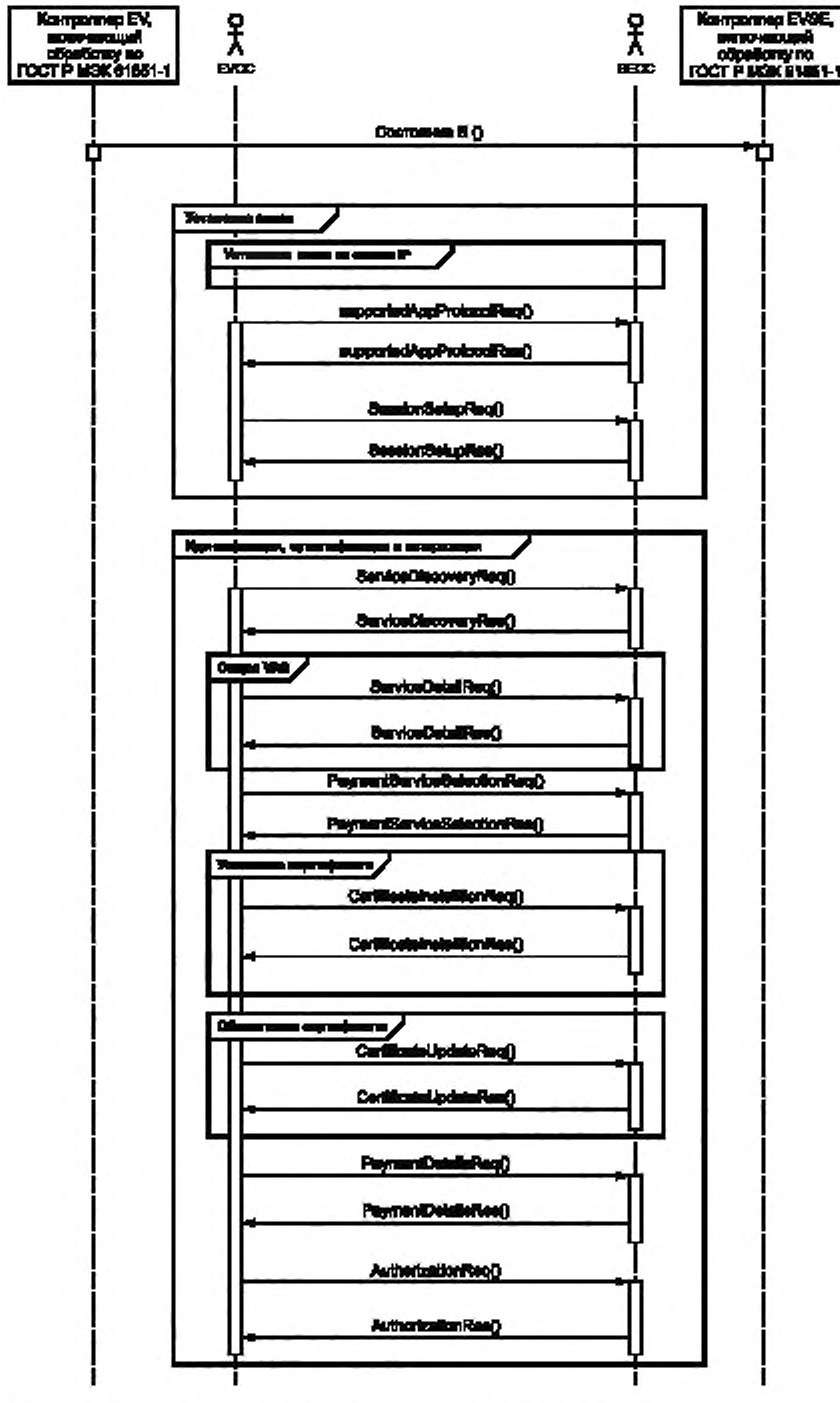


Рисунок 108 — Обзор последовательности сообщений запрос-ответ для постоянного тока, идентификационный режим PnC (1 из 2)

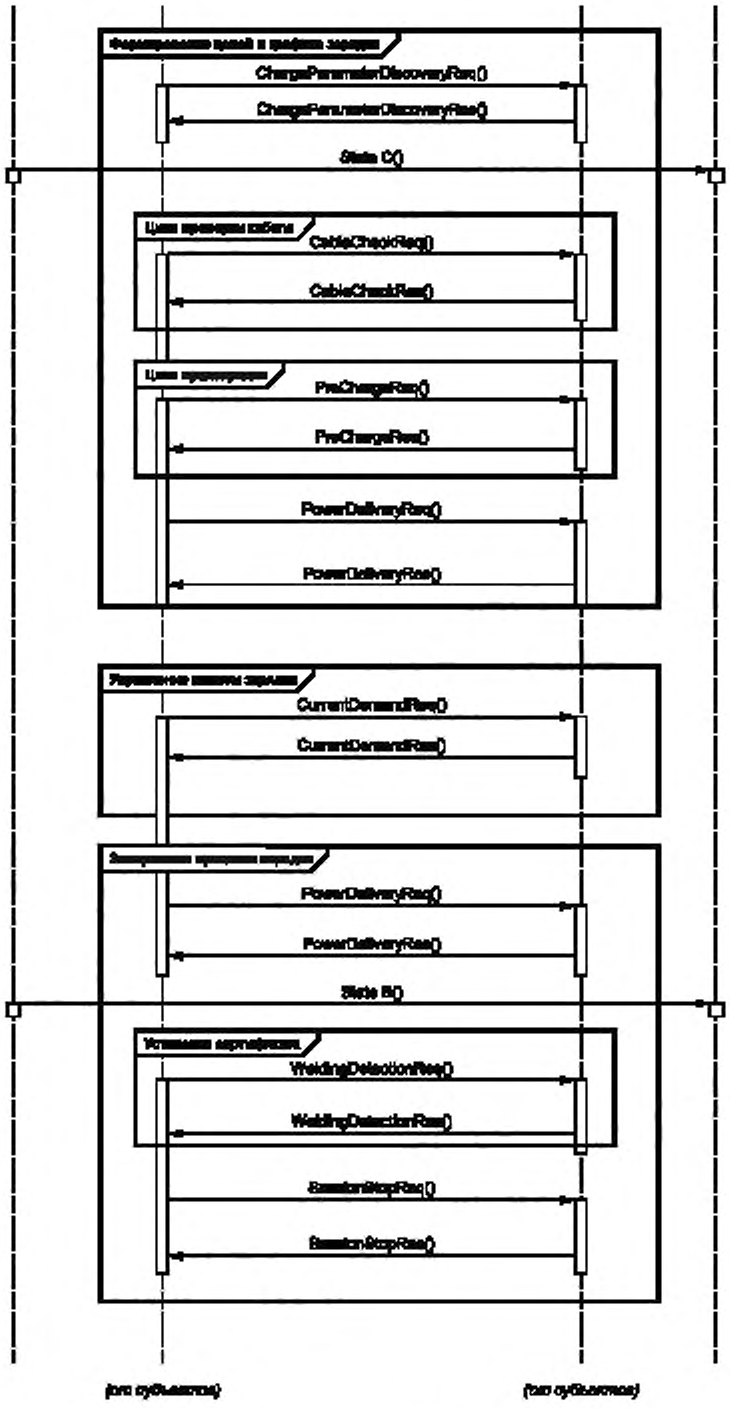


Рисунок 108, лист 2

Приложение А
(справочное)

Сводная таблица требований

В таблице А.1 приведен перечень требований, установленных настоящим стандартом, для удобства их обработки. Данная таблица также используется для ведения истории изменений требований, содержащихся в настоящем стандарте.

Таблица А.1 — Сводные требования

Пункт, включающий требование(я)	Перечень номеров требований
5.3 Использование RFC-ссылок	[V2G2-001], [V2G2-002], [V2G2-003]
7.3.2 Управление сертификатами и ключами	[V2G2-004], [V2G2-005], [V2G2-006], [V2G2-007], [V2G2-885], [V2G2-926], [V2G2-925], [V2G2-886], [V2G2-887], [V2G2-910]
7.3.3 Число корневых сертификатов и срок их действия, глубина и размер сертификата	[V2G2-008], [V2G2-009], [V2G2-010], [V2G2-011], [V2G2-012], [V2G2-877], [V2G2-878], [V2G2-867], [V2G2-869], [V2G2-882], [V2G2-927], [V2G2-883], [V2G2-868], [V2G2-911]
7.3.4 Поддержка и применение TLS	[V2G2-630], [V2G2-631], [V2G2-632], [V2G2-633], [V2G2-634], [V2G2-635], [V2G2-636], [V2G2-637], [V2G2-638], [V2G2-639], [V2G2-640], [V2G2-643], [V2G2-644]
7.4 Состояния связи V2G и работа с каналом данных	[V2G2-014], [V2G2-016], [V2G2-017], [V2G2-018], [V2G2-019], [V2G2-020], [V2G2-021], [V2G2-022], [V2G2-023], [V2G2-024], [V2G2-025], [V2G2-026], [V2G2-027], [V2G2-029], [V2G2-030], [V2G2-031], [V2G2-032], [V2G2-033], [V2G2-034], [V2G2-645], [V2G2-646], [V2G2-717], [V2G2-718], [V2G2-719], [V2G2-720], [V2G2-721], [V2G2-722], [V2G2-723], [V2G2-724], [V2G2-725], [V2G2-726], [V2G2-727]
7.5 Канальный уровень	[V2G2-035], [V2G2-036]
7.6.2.1 IPv6	[V2G2-037], [V2G2-038], [V2G2-039], [V2G2-040], [V2G2-041], [V2G2-042]
7.6.2.2 Протокол динамического управления хостами (DHCPV6)	[V2G2-043], [V2G2-044]
7.6.2.3 Обнаружение соседних объектов (OCO)	[V2G2-045], [V2G2-046]
7.6.2.4 Протокол управления сообщениями в сети Интернет (ICMP)	[V2G2-047], [V2G2-049]
7.6.3.2 Автоконфигурирование адресов без состояния (SLAAC)	[V2G2-050], [V2G2-051], [V2G2-052], [V2G2-053]
7.6.3.3 Выбор адресов	[V2G2-054]
7.7.1.2 Применимые RFC, ограничения и настройки параметров протокола (TCP)	[V2G2-055]
7.7.1.3 Требования к производительности TCP и к контрольной сумме	[V2G2-057], [V2G2-058], [V2G2-059], [V2G2-060], [V2G2-061], [V2G2-062], [V2G2-063], [V2G2-064]
7.7.2.2 Применимые RFC, ограничения и настройки параметров протокола	[V2G2-065]
7.7.3.2 Применимые RFC	[V2G2-067]
7.7.3.3 Применение безопасности транспортного уровня	[V2G2-068], [V2G2-070], [V2G2-649], [V2G2-650], [V2G2-651], [V2G2-870], [V2G2-876], [V2G2-871], [V2G2-872], [V2G2-923], [V2G2-924], [V2G2-873], [V2G2-874], [V2G2-875], [V2G2-810], [V2G2-811]

Продолжение таблицы А.1

Пункт, включающий требование(я)	Перечень номеров требований
7.7.3.4 Удостоверения транспортного уровня и наборы шифров безопасности	[V2G2-071], [V2G2-602], [V2G2-603]
7.8.2 Поддерживаемые порты	[V2G2-073], [V2G2-074], [V2G2-075], [V2G2-076], [V2G2-077], [V2G2-078], [V2G2-079], [V2G2-080], [V2G2-081]
7.8.3.1 Структура PDU V2GTR	[V2G2-082], [V2G2-083], [V2G2-084], [V2G2-085], [V2G2-086], [V2G2-087], [V2G2-088]
7.8.3.2 Обработка заголовка	[V2G2-089], [V2G2-090], [V2G2-092], [V2G2-094], [V2G2-096], [V2G2-800]
7.9.1.1 Обзор	[V2G2-097]
7.9.1.3 Настройки EXI для сообщений прикладного уровня	[V2G2-098], [V2G2-099], [V2G2-100], [V2G2-101], [V2G2-102], [V2G2-600]
7.9.2.2 Удостоверения и наборы шифров безопасности прикладного уровня	[V2G2-103], [V2G2-104]
7.9.2.3 Сертификаты контракта как удостоверения XML-подписи	[V2G2-108]
7.9.2.4.2 Механизм XML-подписи	[V2G2-117], [V2G2-119], [V2G2-121], [V2G2-764], [V2G2-765], [V2G2-766], [V2G2-767], [V2G2-768], [V2G2-769], [V2G2-770], [V2G2-771], [V2G2-909]
7.9.2.4.3 Механизм шифрования	[V2G2-121], [V2G2-122], [V2G2-814], [V2G2-815], [V2G2-816], [V2G2-817], [V2G2-818], [V2G2-819], [V2G2-820], [V2G2-821], [V2G2-822], [V2G2-823]
7.9.2.4.4 Генерирование случайных чисел	[V2G2-835]
7.9.2.4.5 Применение механизмов безопасности к XML-сообщению	[V2G2-652], [V2G2-653]
7.10.1.1 Общая информация	[V2G2-123]
7.10.1.2 Поддерживаемые порты	[V2G2-124], [V2G2-125], [V2G2-126]
7.10.1.3.1 Структура	[V2G2-127], [V2G2-128], [V2G2-129], [V2G2-130], [V2G2-131], [V2G2-132]
7.10.1.3.2 Обработка заголовка	[V2G2-133], [V2G2-134]
7.10.1.4 Сообщение-запрос об обнаружении SECC	[V2G2-135], [V2G2-136], [V2G2-137], [V2G2-138], [V2G2-139], [V2G2-140], [V2G2-141], [V2G2-142], [V2G2-622], [V2G2-623]
7.10.1.5 Сообщение-ответ об обнаружении SECC	[V2G2-143], [V2G2-144], [V2G2-145], [V2G2-146], [V2G2-147], [V2G2-149], [V2G2-150], [V2G2-151], [V2G2-152], [V2G2-153], [V2G2-154], [V2G2-155], [V2G2-156]
7.10.1.6 Хронирование и обработка ошибок	[V2G2-157], [V2G2-158], [V2G2-159], [V2G2-160], [V2G2-161], [V2G2-162]
7.10.1.7 Протокол и обработка опций безопасности	[V2G2-625], [V2G2-626], [V2G2-627], [V2G2-628], [V2G2-629], [V2G2-163], [V2G2-164]
8.1 Общая информация и определения	[V2G2-809]
8.2.1 Последовательность подтверждения	[V2G2-165], [V2G2-166], [V2G2-167], [V2G2-168], [V2G2-169], [V2G2-170], [V2G2-171], [V2G2-172], [V2G2-173], [V2G2-174]
8.2.2 Определение сообщения supported AppProtocolReq и supportedAppProtocolRes	[V2G2-175], [V2G2-176]
8.2.3 Описание семантики сообщений supported AppProtocol	[V2G2-178]

Продолжение таблицы А.1

Пункт, включающий требование(я)	Перечень номеров требований
8.3.2 Определение сообщения	[V2G2-179], [V2G2-180]
8.3.3 Определение заголовка сообщения	[V2G2-181], [V2G2-182]
8.3.4 Определение тела сообщения	[V2G2-183]
8.4.2 Обработка сеанса	[V2G2-739], [V2G2-740], [V2G2-741], [V2G2-742], [V2G2-743], [V2G2-744], [V2G2-745], [V2G2-746], [V2G2-747], [V2G2-748], [V2G2-749], [V2G2-750], [V2G2-751], [V2G2-752], [V2G2-753], [V2G2-754], [V2G2-755], [V2G2-756]
8.4.3.2 SessionSetupReq/Res	[V2G2-188], [V2G2-189], [V2G2-190], [V2G2-191], [V2G2-192], [V2G2-879]
8.4.3.3 ServiceDiscoveryReq/Res	[V2G2-193], [V2G2-194], [V2G2-195], [V2G2-196]
8.4.3.4 ServiceDetailReq/Res	[V2G2-197], [V2G2-198], [V2G2-199], [V2G2-200]
8.4.3.5 PaymentServiceSelectionReq/Res	[V2G2-201], [V2G2-202], [V2G2-203], [V2G2-204]
8.4.3.6 PaymentDetailsReq/Res	[V2G2-205], [V2G2-206], [V2G2-208], [V2G2-209], [V2G2-825], [V2G2-826], [V2G2-898], [V2G2-899]
8.4.3.7 AuthorizationReq/Res	[V2G2-210], [V2G2-211], [V2G2-212], [V2G2-213], [V2G2-697], [V2G2-698], [V2G2-900], [V2G2-901]
8.4.3.8 ChargeParameterDiscoveryReq/Res	[V2G2-214], [V2G2-215], [V2G2-216], [V2G2-218], [V2G2-219], [V2G2-220], [V2G2-761], [V2G2-784], [V2G2-785], [V2G2-786]
8.4.3.9 PowerDeliveryReq/Res	[V2G2-221], [V2G2-222], [V2G2-223], [V2G2-224], [V2G2-225], [V2G2-226], [V2G2-777]], [V2G2-285]
8.4.3.10 CertificateUpdateReq/Res	[V2G2-227], [V2G2-228], [V2G2-229], [V2G2-230], [V2G2-231], [V2G2-232], [V2G2-928], [V2G2-233], [V2G2-696], [V2G2-888], [V2G2-889], [V2G2-827], [V2G2-890], [V2G2-891], [V2G2-892]
8.4.3.11 CertificateInstallationReq/Res	[V2G2-234], [V2G2-235], [V2G2-236], [V2G2-237], [V2G2-238], [V2G2-648], [V2G2-893], [V2G2-894], [V2G2-895], [V2G2-896], [V2G2-897]
8.4.3.12 SessionStopReq/Res	[V2G2-239], [V2G2-240], [V2G2-241], [V2G2-738]
8.4.3.13 MeteringReceiptReq/Res	[V2G2-245], [V2G2-246], [V2G2-247], [V2G2-248], [V2G2-902], [V2G2-903], [V2G2-904], [V2G2-776]
8.4.4.2 ChargingStatusReq/Res	[V2G2-242], [V2G2-243], [V2G2-244]
8.4.5.2 CableCheckReq/Res	[V2G2-249], [V2G2-250], [V2G2-251], [V2G2-252]
8.4.5.3 PreChargeReq/Res	[V2G2-253], [V2G2-254], [V2G2-255], [V2G2-256]
8.4.5.4 CurrentDemandReq/Res	[V2G2-257], [V2G2-258], [V2G2-259], [V2G2-260]
8.4.5.5 WeldingDetectionReq/Res	[V2G2-261], [V2G2-262], [V2G2-263], [V2G2-264]
8.5.2.1 ServiceType	[V2G2-265], [V2G2-266]
8.5.2.2 ServiceListType	[V2G2-267], [V2G2-268]
8.5.2.3 ChargeServiceType	[V2G2-271], [V2G2-272]
8.5.2.4 SupportedEnergyTransferModeType	[V2G2-757], [V2G2-758], [V2G2-759]
8.5.2.5 CertificateChainType	[V2G2-274], [V2G2-275]
8.5.2.6 MeterInfoType	[V2G2-276], [V2G2-277], [V2G2-830], [V2G2-831]
8.5.2.7 PhysicalValueType	[V2G2-278], [V2G2-279], [V2G2-832]

Продолжение таблицы А.1

Пункт, включающий требование(я)	Перечень номеров требований
8.5.2.8 NotificationType	[V2G2-280], [V2G2-281]
8.5.2.9 PaymentOptionListType	[V2G2-282], [V2G2-283]
8.5.2.10 ChargingProfileType	[V2G2-284], [V2G2-606]
8.5.2.11 ProfileEntryType	[V2G2-288], [V2G2-289], [V2G2-290], [V2G2-291], [V2G2-292], [V2G2-293], [V2G2-607], [V2G2-829]
8.5.2.12 SAScheduleListType	[V2G2-294], [V2G2-296], [V2G2-297], [V2G2-298], [V2G2-608]
8.5.2.13 SAScheduleTupleType	[V2G2-299], [V2G2-300], [V2G2-301], [V2G2-303], [V2G2-304], [V2G2-305], [V2G2-306], [V2G2-307], [V2G2-308], [V2G2-309], [V2G2-609], [V2G2-773], [V2G2-905], [V2G2-906], [V2G2-907], [V2G2-908]
8.5.2.14 PMaxScheduleType	[V2G2-310], [V2G2-610]
8.5.2.15 PMaxScheduleEntryType	[V2G2-313], [V2G2-314], [V2G2-315], [V2G2-611]
8.5.2.16 SalesTariffType	[V2G2-316], [V2G2-317], [V2G2-318], [V2G2-612], [V2G2-805]
8.5.2.17 SalesTariffEntryType	[V2G2-321], [V2G2-322], [V2G2-324], [V2G2-325], [V2G2-613], [V2G2-803], [V2G2-802]
8.5.2.18 RelativeTimeIntervalType	[V2G2-327], [V2G2-328], [V2G2-329], [V2G2-330], [V2G2-331], [V2G2-614], [V2G2-833], [V2G2-834]
8.5.2.19 ConsumptionCostType	[V2G2-332], [V2G2-333], [V2G2-334], [V2G2-615]
8.5.2.20 CostType	[V2G2-335], [V2G2-336], [V2G2-337], [V2G2-338], [V2G2-339], [V2G2-340], [V2G2-616], [V2G2-772], [V2G2-775], [V2G2-806], [V2G2-807], [V2G2-808]
8.5.2.21 ServiceParameterListType	[V2G2-343], [V2G2-344]
8.5.2.22 ParameterSetType	[V2G2-345], [V2G2-346]
8.5.2.23 ParameterType	[V2G2-347], [V2G2-348]
8.5.2.24 SelectedServiceListType	[V2G2-349], [V2G2-350]
8.5.2.25 SelectedServiceType	[V2G2-351], [V2G2-352]
8.5.2.26 SubCertificatesType	[V2G2-353], [V2G2-354], [V2G2-656]
8.5.2.27 ListOfRootCertificateIDsType	[V2G2-355], [V2G2-356]
8.5.2.28 ContractSignatureEncryptedPrivateKeyType	[V2G2-778], [V2G2-779]
8.5.2.29 DiffieHellmanPublickeyType	[V2G2-780], [V2G2-781]
8.5.2.30 EMAIDType	[V2G2-782], [V2G2-783]
8.5.3.1 AC_EVSEStatusType	[V2G2-358], [V2G2-359]
8.5.3.2 AC_EVChargeParameterType	[V2G2-360], [V2G2-361]
8.5.3.3 AC_EVSEChargeParameterType	[V2G2-362], [V2G2-363,] [V2G2-708], [V2G2-709]
8.5.4.1 DC_EVSEStatusType	[V2G2-364], [V2G2-365], [V2G2-366], [V2G2-801]
8.5.4.2 DC_EVStatusType	[V2G2-367], [V2G2-368], [V2G2-369]
8.5.4.3 DC_EVChargeParameterType	[V2G2-370], [V2G2-371]
8.5.4.4 DC_EVSEChargeParameterType	[V2G2-372], [V2G2-373]
8.5.4.5 DC_EVPowerDeliveryParameterType	[V2G2-374], [V2G2-375]

Продолжение таблицы А.1

Пункт, включающий требование(я)	Перечень номеров требований
8.6.1 Обзор	[V2G2-760], [V2G2-828]
8.6.2.1 Обзор поддерживаемых наборов сообщений	[V2G2-659], [V2G2-660], [V2G2-661], [V2G2-662], [V2G2-663], [V2G2-664], [V2G2-665], [V2G2-666], [V2G2-667], [V2G2-668]
8.6.2.2 Общие требования	[V2G2-762], [V2G2-763]
8.6.2.3 Переменный ток	[V2G2-376], [V2G2-377], [V2G2-378], [V2G2-379], [V2G2-380], [V2G2-381], [V2G2-384], [V2G2-385], [V2G2-386], [V2G2-387], [V2G2-388], [V2G2-389]
8.6.2.4 Постоянный ток	[V2G2-390], [V2G2-391], [V2G2-392], [V2G2-393], [V2G2-394], [V2G2-395], [V2G2-396], [V2G2-397], [V2G2-398], [V2G2-399], [V2G2-400], [V2G2-401]
8.6.3.1 Наборы сообщений для зарядки переменным/постоянным током в режимах EIM/PnC	[V2G2-402], [V2G2-403], [V2G2-404], [V2G2-405]
8.6.3.2 Набор сообщений квитанции учета	[V2G2-406], [V2G2-407], [V2G2-408], [V2G2-691], [V2G2-787], [V2G2-788], [V2G2-789]
8.6.3.3 Установка сертификата	[V2G2-410], [V2G2-411]
8.6.3.4 Обновление сертификата	[V2G2-412], [V2G2-413]
8.6.3.5 Набор сообщений дополнительных услуг	[V2G2-414], [V2G2-415]
8.6.3.6 Выбор сервисов	[V2G2-416], [V2G2-417], [V2G2-418], [V2G2-419], [V2G2-420], [V2G2-421], [V2G2-422], [V2G2-424], [V2G2-425], [V2G2-426], [V2G2-427], [V2G2-428], [V2G2-429], [V2G2-430], [V2G2-431], [V2G2-432], [V2G2-433], [V2G2-774]
8.7.2.1 Определения	[V2G2-434], [V2G2-435]
8.7.2.2 Хронирование EVCC пар сообщений запрос-ответ	[V2G2-436], [V2G2-437], [V2G2-438], [V2G2-439], [V2G2-440]
8.7.2.3 Хронирование SECC последовательности сообщений запрос-ответ	[V2G2-441], [V2G2-442], [V2G2-443], [V2G2-444], V2G2-445]
8.7.3.1 Определения	[V2G2-605]
8.7.3.2 Хронирование EVCC для установления сеанса связи	[V2G2-446], [V2G2-447], [V2G2-448], [V2G2-449]
8.7.3.3 Хронирование SECC для установления сеанса связи	[V2G2-714], [V2G2-715], [V2G2-716]
8.7.3.4 Хронирование EVCC для параметра EVSEProcessing	[V2G2-710], [V2G2-711]
8.7.3.5 Хронирование SECC для параметра EVSEProcessing	[V2G2-712], [V2G2-713]
8.7.3.6 Хронирование EVCC для проверки кабеля	[V2G2-700], [V2G2-701], [V2G2-702], [V2G2-703]
8.7.3.7 Хронирование EVCC для предзарядки	[V2G2-704], [V2G2-705], [V2G2-706], [V2G2-707]
8.7.4.2 Общие требования	[V2G2-836], [V2G2-837], [V2G2-838], [V2G2-839], [V2G2-840], [V2G2-841], [V2G2-842], [V2G2-843], [V2G2-844], [V2G2-845], [V2G2-850], [V2G2-854], [V2G2-855], [V2G2-856], [V2G2-731], [V2G2-737], [V2G2-733]
8.7.4.3 Специальные требования для переменного тока	[V2G2-846], [V2G2-847], [V2G2-848], [V2G2-849], [V2G2-851], [V2G2-852], [V2G2-853], [V2G2-857], [V2G2-858], [V2G2-859], [V2G2-860], [V2G2-861], [V2G2-862], [V2G2-863], [V2G2-864], [V2G2-865], [V2G2-866], [V2G2-930], [V2G2-931]

Окончание таблицы А.1

Пункт, включающий требование(я)	Перечень номеров требований
8.7.4.4 Специальные требования для постоянного тока	[V2G2-912], [V2G2-913], [V2G2-914], [V2G2-915], [V2G2-916], [V2G2-917], [V2G2-918], [V2G2-919], [V2G2-920], [V2G2-921], [V2G2-922]
8.8.3 Обработка ResponseCode	[V2G2-457], [V2G2-458], [V2G2-459], [V2G2-460], [V2G2-461], [V2G2-462], [V2G2-463], [V2G2-464], [V2G2-465], [V2G2-467], [V2G2-468], [V2G2-469], [V2G2-470], [V2G2-471], [V2G2-472], [V2G2-473], [V2G2-474], [V2G2-475], [V2G2-476], [V2G2-477], [V2G2-478], [V2G2-479], [V2G2-480], [V2G2-481], [V2G2-690], [V2G2-693], [V2G2-694], [V2G2-695], [V2G2-734], [V2G2-735], [V2G2-736], [V2G2-804], [V2G2-824]
8.8.4.1 Общие положения	[V2G2-672], [V2G2-673], [V2G2-674], [V2G2-675], [V2G2-676], [V2G2-677], [V2G2-678], [V2G2-679], [V2G2-680]
8.8.4.2 EVCC	[V2G2-482], [V2G2-483], [V2G2-484], [V2G2-485], [V2G2-486], [V2G2-487], [V2G2-488], [V2G2-489], [V2G2-490], [V2G2-491], [V2G2-492], [V2G2-493], [V2G2-494], [V2G2-495], [V2G2-496], [V2G2-497], [V2G2-498], [V2G2-499], [V2G2-500], [V2G2-501], [V2G2-502], [V2G2-503], [V2G2-504], [V2G2-505], [V2G2-506], [V2G2-507], [V2G2-508], [V2G2-510], [V2G2-511], [V2G2-512], [V2G2-513], [V2G2-514], [V2G2-516], [V2G2-517], [V2G2-518], [V2G2-519], [V2G2-520], [V2G2-521], [V2G2-522], [V2G2-524], [V2G2-525], [V2G2-526], [V2G2-527], [V2G2-528], [V2G2-530], [V2G2-531], [V2G2-532], [V2G2-533], [V2G2-534], [V2G2-535], [V2G2-617], [V2G2-618], [V2G2-619], [V2G2-620], [V2G2-683], [V2G2-684], [V2G2-685], [V2G2-686], [V2G2-689], [V2G2-728], [V2G2-790], [V2G2-791], [V2G2-792], [V2G2-793], [V2G2-799], [V2G2-880], [V2G2-599]
8.8.4.3 SECC	[V2G2-536], [V2G2-537], [V2G2-538], [V2G2-539], [V2G2-540], [V2G2-541], [V2G2-542], [V2G2-543], [V2G2-544], [V2G2-545], [V2G2-546], [V2G2-547], [V2G2-548], [V2G2-549], [V2G2-550], [V2G2-551], [V2G2-552], [V2G2-553], [V2G2-554], [V2G2-555], [V2G2-556], [V2G2-557], [V2G2-558], [V2G2-559], [V2G2-560], [V2G2-561], [V2G2-562], [V2G2-563], [V2G2-564], [V2G2-566], [V2G2-567], [V2G2-568], [V2G2-569], [V2G2-570], [V2G2-571], [V2G2-572], [V2G2-573], [V2G2-574], [V2G2-575], [V2G2-576], [V2G2-577], [V2G2-578], [V2G2-579], [V2G2-580], [V2G2-581], [V2G2-582], [V2G2-583], [V2G2-584], [V2G2-585], [V2G2-586], [V2G2-587], [V2G2-588], [V2G2-589], [V2G2-590], [V2G2-592], [V2G2-593], [V2G2-595], [V2G2-596], [V2G2-597], [V2G2-598], [V2G2-601], [V2G2-621], [V2G2-687], [V2G2-688], [V2G2-729], [V2G2-812], [V2G2-813], [V2G2-881], [V2G2-795], [V2G2-796], [V2G2-797], [V2G2-798]
Приложение В Профили сертификатов	[V2G2-923], [V2G2-933]

Приложение В
(обязательное)

Профили сертификатов

[V2G2-884] — каждый сертификат, используемый в настоящем стандарте, должен соответствовать надлежащему профилю, указанному в настоящем приложении, т. е. в таблицах В.1—В.6.

Профили сертификатов, представленные в таблицах, делятся на несколько тематических групп, каждая из которых содержит профили для конкретных типов сертификатов. В каждой группе корневой сертификат должен быть действительным корневым сертификатом V2G, если не указано иное.

Содержание ячеек имеет следующее значение:

x Требуется

x) Дополнительно

— Не должно присутствовать

c Это расширение **критичное**, см. *[4]*. Если реализация распознает присутствие «критичного» расширения, но не может интерпретировать его, то реализация должна отклонить сертификат

nc Это расширение **некритичное**, см. *[4]*. Если реализация распознает присутствие «некритичного» расширения, но реализация не может интерпретировать его, то данное расширение может быть проигнорировано. Поэтому все дополнительные поля также должны быть «некритичными».
Выдержка из *[4]*: «Использующая сертификаты система ДОЛЖНА отклонить сертификат, если она сталкивается с критичным расширением, которое она не распознает, или с критичным расширением, содержащим информацию, которую она не может обработать. Некритичное расширение МОЖЕТ быть проигнорировано, если оно не распознается, но ДОЛЖНО быть обработано, если оно распознается»

PCID Идентификатор сервисного сертификата (макс. 17)

CPID Идентификатор зарядной точки

PEID Идентификатор частной среды

Примечание 1 — Применяются следующие идентификаторы объектов: {iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 2} и {iso(1) member-body(2) us(840) ansi-X9-62(10045) curves(3) prime(1) prime256v1(7)}.

Примечание 2 — Протокол статуса онлайнового сертификата (OCSP), описанный в *[5]*, является протоколом, используемым для получения статуса отзыва сертификата x.509. OCSP является альтернативой спискам аннулированных сертификатов (COC), а также является онлайновым сервисом. Это означает, что инфраструктура серверной системы должна поддерживать услуги OCSP для того, чтобы быть в состоянии пользоваться этим сервисом. Поскольку доступ к услугам OCSP не может быть гарантирован во время зарядки, использование OCSP может быть только рекомендуемым, но не обязательным.

В сертификате x.509 идентификатор объектов id-ad-ocsp используется при санкционированном доступе к информации в качестве метода доступа, если информация об отзыве для сертификата, содержащего данное расширение, доступна с помощью протокола статуса онлайнового сертификата (OCSP). Когда в качестве метода доступа фигурирует id-ad-ocsp, поле accessLocation является адресом отвечающего по OCSP с использованием понятий, описанных в *[5]*.

Таблица В.1 — Корневой сертификат V2G

Группа:		V2G корневой
Имя:		V2G корневой
Тип:		Корневой
tbsCertificate	Версия	2 (X.509v3)
	Серийный номер	Число
	Подпись	ecdsa-c-SHA256
Издатель	Страна	(x)
	Организация	x
	Подразделение организации	(x)
	Стандартное имя	x

Окончание таблицы В.1

Срок действия		40 лет
Субъект	Страна	(x)
	Организация	x
	Подразделение организации	(x)
	Стандартное имя	x
	Компонент домена	"V2G"
Информация об открытом ключе субъекта	Открытый ключ	x
	Криптографический алгоритм	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc
	Идентификатор ключа субъекта	(x)/nc
	Использование ключа:	c
	цифровая подпись	—
	невозможность отказа (обязательство по содержанию)	—
	шифрование ключа	—
	шифрование данных	0
	согласование ключа	
	подписание сертификата ключа	(x)
	Подписание COC	(x)
	Только шифрование	—
	Только дешифрование	—
	Расширенное использование ключа	—
	Полисы сертификатов	(x)/nc
	Базовые ограничения	c
	CA	true
	Длина пути доступа	—
	Точки рассылки COC	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/адрес отвечающего по OCSP
Значение подписи	Криптографический алгоритм	ecdsa-c-SHA256
		Строка октетов

Таблица В.2 — Сертификаты оператора зарядной точки (ОЗТ)

Группа:		Оператор зарядной точки (ОЗТ)			OCSP
Имя:		ОЗТ Суб 1	ОЗТ Суб 2	Серт. SECC	Серт. OCSP
Тип:		Суб	Суб	Листовой	Листовой

Продолжение таблицы В.2

tbsCertificate	Версия	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)
	Серийный номер	Целое число	Целое число	Целое число	Целое число
	Подпись	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Издатель	Страна	(x)	(x)	(x)	(x)
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	x	x
Срок действия		4 года	1—2 года	2—3 месяца	до 1 года
Субъект	Страна	(x)	(x)	x	—
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	CPID	x
	Компонент домена	(x)	(x)	"CPO"	(x)
Информация об открытом ключе субъекта	Открытый ключ	x	x	x	x
	Криптографический алгоритм	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Идентификатор ключа субъекта	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Использование ключа:	c	c	c	c
	цифровая подпись	—	—	x	—
	невозможность отказа (обязательство по содержанию)	—	—	—	—
	шифрование ключа	—	—	—	—
	шифрование данных	0	0	0	0
	согласование ключа	—	—	—	—
	подписание сертификата ключа	(x)	(x)	—	—
	Подписание СОС	(x)	(x)	—	—
	Только шифрование	—	—	—	—

Окончание таблицы В.2

Расширения	Только дешифрование	—	—	—	—
	Расширенное использование ключа	—	—	—	1.3.6.1.5.5.7.3.9 — OCSPSigning
	Полисы сертификатов	—	—	—	—
	Базовые ограничения	c	c	c	c
	CA	true	true	false	false
	Длина пути доступа	1	0	—	—
	Точки рассылки COC	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/ адрес ответвающего по OCSP	(x)/nc id-ad-ocsp/ адрес ответвающего по OCSP	(x)/nc id-ad-ocsp/ адрес ответвающего по OCSP	(x)/nc id-ad-ocsp/ адрес ответвающего по OCSP
	Криптографический алгоритм	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Значение подписи		Строка октетов	Строка октетов	Строка октетов	Строка октетов

Таблица В.3 — Сертификаты сервиса по установке сертификатов (сервисные)

Группа:		Сервис по установке сертификатов (Серв.)		
Имя:		Серв. Суб 1	Серв. Суб 2	Серв. Сервис
Тип:		Суб	Суб	Листовой
tbsCertificate	Версия	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)
	Серийный номер	Целое число	Целое число	Целое число
	Подпись	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Издатель	Страна	(x)	(x)	(x)
	Организация	x	x	x
	Подразделение организации	(x)	(x)	(x)
	Стандартное имя	x	x	x
Срок действия		4 года	1—2 года	2—3 месяца
Субъект	Страна	(x)	(x)	x
	Организация	x	x	x
	Подразделение организации	(x)	(x)	(x)
	Стандартное имя	x	x	x
	Компонент домена	(x)	(x)	"CPS"

Окончание таблицы В.3

Информация об открытом ключе субъекта	Открытый ключ	x	x	x
	Криптографический алгоритм	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc	(x)/nc	(x)/nc
	Идентификатор ключа субъекта	(x)/nc	(x)/nc	(x)/nc
	Использование ключа:	c	c	c
	цифровая подпись	—	—	x
	невозможность отказа (обязательство по содержанию)	—	—	—
	шифрование ключа	—	—	—
	шифрование данных	0	0	0
	согласование ключа	—	—	—
	подписание сертификата ключа	(x)	(x)	—
	Подписание СОС	(x)	(x)	—
	Только шифрование	—	—	—
	Только дешифрование	—	—	—
	Расширенное использование ключа	—	—	—
	Полисы сертификатов	—	—	—
	Базовые ограничения	c	c	c
	CA	true	true	false
	Длина пути доступа	1	0	—
	Точки рассылки СОС	(x)/nc	(x)/nc	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP
	Криптографический алгоритм	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Значение подписи		Строка октетов	Строка октетов	Строка октетов

Таблица В.4 — Сертификаты оператора услуг для EV

Группа:		Оператор услуг для EV			
Имя:		Корневой СА ОУЭТС	Суб 1 ОУЭТС	Суб 2 ОУЭТС	Серт. контракта
Тип:		Корневой	Суб	Суб/Листовой	Листовой
tbsCertificate	Версия	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)
	Серийный номер	Целое число	Целое число	Целое число	Целое число
	Подпись	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Издатель	Страна	(x)	(x)	(x)	(x)
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	x	x
Срок действия		[на усмотрение оператора услуг для EV]	[на усмотрение оператора услуг для EV]	[на усмотрение оператора услуг для EV]	4 недели — 2 года
Субъект	Страна	(x)	(x)	(x)	—
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	x	ИАЭТС
	Компонент домена	(x)	(x)	(x)	(x)
Информация об открытом ключе субъекта	Открытый ключ	x	x	x	x
	Криптографический алгоритм	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Идентификатор ключа субъекта	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Использование ключа:	c	c	c	c
	цифровая подпись			x	x
	невозможность отказа (обязательство по содержанию)	—	—	x	x
	шифрование ключа	—	—	—	x
	шифрование данных	0	0	0	0
	согласование ключа	—	—	—	x

Окончание таблицы В.4

Расширения	подписание сертификата ключа	(x)	(x)	(x)	—
	Подписание COC	(x)	(x)	(x)	—
	Только шифрование	—	—	—	—
	Только дешифрование	—	—	—	—
	Расширенное использование ключа	—	—	—	—
	Полисы сертификатов	(x)/nc	—	—	—
	Базовые ограничения	c	c	c	c
	CA	true	true	true	false
	Длина пути доступа	—	1	0	—
	Точки рассылки COC	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP
	Криптографический алгоритм	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Значение подписи		Строка октетов	Строка октетов	Строка октетов	Строка октетов

Примечание 1 — СубУО 2 операторов услуг для EV имеет две задачи: во-первых, он удостоверяет сертификаты контрактов; во-вторых, он подписывает тарифную информацию, предназначенную для владельцев этих сертификатов контрактов.

Примечание 2 — Идентификатор сервисного сертификата (CertID) используется оператором услуг для EV с целью идентификации контракта, принадлежащего данному EV. Это возможно, потому что оператором услуг для EV при оформлении контракта потребителю был дан CertID (CN и O, см. ниже). Для этой цели CertID должен быть короткой цепочкой, уникальной для изготовителя, создавшего сервисный сертификат. CertID содержится в сервисном сертификате.

[V2G2-932] CertID должен быть буквенно-цифровой строкой с максимальной длиной 18 знаков (т. е. A...Z, a...z, 0...9).

[V2G2-933] CertID должен содержаться в поле субъекта сервисного сертификата следующим образом:

- сам CertID (см. [V2G2-932]) должен быть значением стандартного имени (CN), отличительного от имени (DN);
- имя изготовителя должно кодироваться в поле Organization (O) с использованием уникального идентификатора, выбранного изготовителем для своей идентификации. Вместе с CN он образует уникальное имя;
- имя X.500 в поле субъекта не должно содержать какие-либо дополнительные значения.

Таблица В.5 — Сертификаты изготовителя EV

Группа:		Изготовитель			
Имя:		Корневой CA изготовителя	Суб 1 изготовителя	Суб 2 изготовителя	Серт. изготовителя

Продолжение таблицы В.5

Тип:		Корневой	Суб	Суб	Листовой
tbsCertificate	Версия	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)	2 (X.509v3)
	Серийный номер	Целое число	Целое число	Целое число	Целое число
	Подпись	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Издатель	Страна	(x)	(x)	(x)	(x)
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	x	x
Срок действия		[на усмотрение изготовителя]	[на усмотрение изготовителя]	[на усмотрение изготовителя]	[на усмотрение изготовителя]
Субъект	Страна	(x)	(x)	(x)	
	Организация	x	x	x	x
	Подразделение организации	(x)	(x)	(x)	(x)
	Стандартное имя	x	x	x	PCID
	Компонент домена	(x)	(x)	(x)	"OEM"
Информация об открытом ключе субъекта	Открытый ключ	x	x	x	x
	Криптографический алгоритм	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Идентификатор ключа субъекта	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Использование ключа:	c	c	c	c
	цифровая подпись	—	—	—	x
	невозможность отказа (обязательство по содержанию)	—	—	—	—
	шифрование ключа	—	—	—	—
	шифрование данных	0	0	0	0
	согласование ключа	—	—	—	x
	подписание сертификата ключа	(x)	(x)	(x)	—

Окончание таблицы В.5

Расширения	Подписание COC	(x)	(x)	(x)	—
	Только шифрование	—	—	—	—
	Только дешифрование	—	—	—	—
	Расширенное использование ключа	—	—	—	—
	Полисы сертификатов	(x)/nc	—	—	—
	Базовые ограничения	c	c	c	c
	CA	true	true	true	false
	Длина пути	—	1	0	—
	Точки рассылки COC	(x)/nc	(x)/nc	(x)/nc	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/ адрес отвечающего по OCSP
	Криптографический алгоритм	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256	ecdsa-c-SHA256
Значение подписи		Строка октетов	Строка октетов	Строка октетов	Строка октетов

Таблица В.6 — Сертификаты для частной среды

Группа:		Настенное зарядное устройство/частная среда	
Имя:		Корневой частной среды	Серт. SECC частной среды
Тип:		Корневой	Листовой
tbsCertificate	Версия	2 (X.509v3)	2 (X.509v3)
	Серийный номер	Целое число	Целое число
	Подпись	ecdsa-c-SHA256	ecdsa-c-SHA256
Издатель	Страна	(x)	(x)
	Организация	x	x
	Подразделение организации	(x)	(x)
	Стандартное имя	x	x
Срок действия		40 лет	2—20 лет
Субъект	Страна	(x)	x
	Организация	x	x
	Подразделение организации	(x)	(x)

Окончание таблицы В.6

Субъект	Стандартное имя	x	PEID
	Компонент домена	(x)	(x)
Информация об открытом ключе субъекта	Открытый ключ	x	x
	Криптографический алгоритм	id-ecPublicKey	id-ecPublicKey
	Параметры	ECParameters (namedCurve secp256r1)	ECParameters (namedCurve secp256r1)
Расширения	Идентификатор ключа ведомства	(x)/nc	(x)/nc
	Идентификатор ключа субъекта	(x)/nc	(x)/nc
	Использование ключа:	c	c
	цифровая подпись	—	x
	невозможность отказа (обязательство по содержанию)	—	—
	шифрование ключа	—	—
	шифрование данных	0	0
	согласование ключа	—	—
	подписание сертификата ключа	(x)	—
	Подписание COC	(x)	—
	Только шифрование	—	—
	Только дешифрование	—	—
	Расширенное использование ключа	—	—
	Полисы сертификатов	—	—
	Базовые ограничения	c	c
	CA	true	false
	Длина пути	0	—
	Точки рассылки COC	(x)/nc	(x)/nc
	Ведомственный доступ к информации (OCSP)	(x)/nc id-ad-ocsp/адрес отвечающего по OCSP	(x)/nc id-ad-ocsp/адрес отвечающего по OCSP
	Криптографический алгоритм	ecdsa-c-SHA256	ecdsa-c-SHA256
Значение подписи		Строка октетов	Строка октетов

Приложение С
(справочное)

Применение сертификатов

С.1 Общая информация

С.1.1 Обзор

Настоящее приложение дает обзорное описание применения сертификатов и требований, которые должны выполняться в соответствии с концепцией сертификатов, реализуемой в настоящем стандарте. В нем также описываются параметры сертификатов, такие как действительность, размер, длина цепочки и другие. Настоящее приложение не представляет собой дополнительные требования.

В протоколе зарядки используются следующие типы сертификатов:

- корневые сертификаты V2G: это глобально действующие корневые сертификаты (высшего уровня) инфраструктуры открытых ключей. Они используются для проверки аутентичности сертификатов. Соответствующие закрытые ключи находятся у соответствующих корневых CA (удостоверяющих органов);
- корневой сертификат оператора услуг для EV: данный вид сертификата используется для подписания сертификатов контракта (через цепочку Sub-CA);
- сертификат контракта: данный вид сертификата используется для случая использования «Plug and Charge» (т. е. при зарядке на основе контракта) для указания на существование контракта между транспортным средством и вторичным субъектом (оператор услуг для EV). Он хранится в EVCC вместе с соответствующим закрытым ключом. EVCC использует его в качестве доказательства существования соответствующего контракта для EVSE. Сертификаты контракта являются производными от корневого сертификата оператора услуг для EV;
- сертификат SECC: данный вид сертификата используется, чтобы аутентифицировать SECC для EVCC. Соответствующий закрытый ключ является принадлежностью SECC. Сертификаты SECC являются производными от вышеупомянутых корневых сертификатов V2G;
- корневой сертификат частного оператора: корневой сертификат частного оператора очень схож с корневым сертификатом оператора услуг для EV. Единственное отличие заключается в том, что его обработка организована по-другому (описано в С.2). Он служит для облегчения создания частной зарядной инфраструктуры для транспортных средств, которые находятся под непосредственным контролем владельца инфраструктуры;
- сервисный сертификат изготовителя: данный вид сертификата индивидуален для каждого транспортного средства (устанавливается, например, при производстве транспортного средства) и используется для проверки идентификации транспортного средства в начале процесса выдачи;
- корневой сертификат изготовителя: данный вид сертификата используется для подписания сервисных сертификатов изготовителя. Каждый изготовитель может создать корневой сертификат (высшего уровня) и рассылать его вторичным субъектам. Корневой сертификат изготовителя не является частью глобальной инфраструктуры открытых ключей, т. е. он не подписывается в обязательном порядке корневым сертификатом V2G.

Далее описываются общие требования и ограничения, вводимые изготовителями (см. С.1.2) и вторичными субъектами (см. С.1.3) в отношении указанных видов сертификатов. Данный пункт используется для обоснования необходимости или, по крайней мере, целесообразности решений, изложенных в настоящем стандарте (см. С.1.4). В заключение дается обзор итоговой структуры и использования сертификатов (см. С.1.5).

С.1.2 Требования изготовителя EV

Изготовитель, как правило, имеет следующие общие требования, объясняющиеся его стремлением защитить блоки управления (в данном случае SECC) от чрезмерного удорожания. Ручная наладка блоков управления (например, на станции техобслуживания) требует больших усилий и ее следует избегать.

R1 — Установка сертификата на транспортное средство является несложной только при производстве транспортного средства. Позднее на станции техобслуживания установка становится гораздо более трудоемкой. Для обеспечения установки сертификата только при производстве транспортного средства сертификат должен быть «статическим». Это означает, что сертификату требуется очень большой срок действия. Поскольку транспортные средства эксплуатируются в течение 20 лет или больше, сертификату требуется еще больший срок действия. Такие статические сертификаты, например, могут быть корневыми сертификатами PKI (инфраструктуры открытых ключей), которые хранятся на транспортном средстве.

R2 — Статические сертификаты не могут использоваться для всех целей. Срок действия сертификата контракта (используемого для режима «Plug and Charge»), как правило, равен сроку действия контракта. Кроме того, контракт не может существовать во время производства транспортного средства, и, как следствие, сертификат контракта также не существует. Должна существовать возможность установки «нестатических» сертификатов на транспортное средство приемлемым методом. В идеале установка сертификата происходит автоматически посредством протокола зарядки. Если это невозможно по какой-либо причине, сертификат может быть отправлен от вторичного субъекта в виде файла и должен

быть установлен на транспортное средство с использованием онлайнового соединения или диагностического интерфейса (на станции техобслуживания). Поэтому преобразований формата следует избегать для снижения затрат и единообразия гарантии применительно ко всем вторичным субъектам. Таким образом, для файлов сертификатов требуется стандартизованный формат файлов (особенно для сертификатов контракта, потому что они не могут быть статическими).

R3 — Блоки управления с большой (постоянной) памятью являются дорогостоящими, особенно если запоминающее устройство должно обеспечивать большое число циклов записи. Для уменьшения общего объема памяти, например энергонезависимой флэш-памяти, память, требующаяся для сертификатов, должна быть небольшой. Отсюда вытекают следующие требования: (R3a) — размер одного сертификата должен быть небольшим. (R3b) — каждый раз, когда EVCC необходимо сохранить цепочку сертификата, длина цепочки должна быть короткой. (R3c) — каждый раз, когда в EVCC необходимо сохранить несколько сертификатов одного и того же типа (например, несколько корневых сертификатов), число таких сертификатов должно быть небольшим.

С.1.3 Требования вторичных субъектов

Вторичный субъект или организация, управляющие своей PKI (инфраструктурой открытых ключей), обычно имеют общие требования, связанные с тем фактом, что расходы на управление PKI должны быть небольшими. Такие расходы увеличиваются, когда требуется координация действий нескольких компаний или организаций либо когда задача не может быть передана нескольким организациям вследствие структуры сертификатов, установленной настоящим стандартом.

R4 — Чтобы иметь возможность рассылать общий корневой сертификат (который устанавливается, например, на каждом транспортном средстве), вторичные субъекты должны создать группу, которая использует общий (т. е. единый) корневой сертификат. Все сертификаты, созданные данными вторичными субъектами, являются производными (опосредованно) от данного корневого сертификата (т. е. сертификаты подписываются закрытым ключом, принадлежащим данному корневому сертификату). Трудно добиться работы всех вторичных субъектов (в глобальном масштабе) только в рамках одной группы. Для вторичных субъектов разных континентов или разных видов (например, операторов EVSE, операторов сети, операторов услуг для EV), по всей видимости, требуется наличие нескольких групп. В случае создания разных групп облегчается их организация (с индивидуальным корневым сертификатом у каждой группы).

R5 — Центральная организация каждой группы создает свой корневой сертификат и соответствующий (очень секретный) закрытый ключ. Маловероятно, что эта организация создаст (и подпишет) все сертификаты, которые требуются всем вторичным субъектам (например, все сертификаты контрактов всех потребителей). Вместо этого каждому вторичному субъекту необходимо наличие возможности создавать свои сертификаты самостоятельно. Для этой цели ему нужен «промежуточный сертификат», который подписывается корневым сертификатом и используется для подписания (листовых) сертификатов, созданных данным вторичным субъектом. Это означает, что необходима цепочка сертификатов. Кроме того, маловероятно, что данная центральная (корневая) организация будет непосредственно подписывать «промежуточные сертификаты» всех вторичных субъектов. Вместо этого необходимы «промежуточные организации» (например, одна для каждой страны). Это приводит к цепочке сертификатов с двумя «промежуточными сертификатами». Если используется большее число «промежуточных сертификатов» (т. е. уровней для промежуточных организаций), становится легче управлять взаимодействием между разными организациями.

R6 — Коммуникация между SECC и информационной системой SA приводит к затратам на связь. Кроме того, такая коммуникация задерживает начало процедуры зарядки и сокращает готовность EVSE (поскольку связь может нарушиться). В связи с этим такой коммуникации следует по возможности избегать (с позиции вторичного субъекта). В оптимальном варианте EVSE должно оставаться в режиме офлайн в течение всей процедуры зарядки.

Примечание — Требование R6 учитывает затраты на связь и задержки до начала зарядки. Однако могут быть также другие приоритеты участников. Кроме того, могут существовать сценарии, где затраты на связь фактически не возникают (например, если EVSE имеет LAN- или WLAN-соединение с Интернетом или его серверной системой). Например, в случае использования [C2] в соответствии с частью 1 настоящего стандарта EVCC осуществляет коммуникацию непосредственно со вторичным субъектом, используя прямой канал TLS. Если предусматривается постоянное наличие канала связи во время зарядки, требование R6 не применяется. Другой пример: SECC может предлагать соединение с Интернетом или TCP/IP-соединение с произвольно выбираемым сервером в качестве (дополнительной) услуги для транспортного средства. Это соединение может существовать постоянно во время зарядки (или даже дольше) и приводит также к коммуникационному трафику. В обоих примерах реализация SECC должна надлежащим образом обеспечивать коммуникационный трафик.

С.1.4 Обоснование решений настоящего стандарта

С учетом требований, изложенных в С.1.2 и С.1.3, далее представляются и поясняются решения, принятые в настоящем стандарте. Поскольку некоторые требования приводят к конфликтующим целям, отдельные решения представляют собой компромисс между требованиями изготовителей и требованиями вторичных субъектов.

D1 — Размер одного сертификата: вследствие требования R3a размер сертификата ограничен. В соответствии с **[V2G2-010]** размер сертификата в DER-кодированной форме должен быть не больше 800 байтов. Это может быть обеспечено выдающим сертификат вторичным субъектом, если соответствующая информация не включена в сертификат (например, отсутствует адрес выдавшего субъекта).

D2 — Длина цепочек сертификата: вследствие требования R3b полные сертификаты, включая их цепочки, должны быть небольшими. Требование R5, однако, предусматривает длинные цепочки, поскольку ими легче управлять. В качестве компромисса **[V2G2-009]** устанавливает, что длина пути доступа ограничивается 3 (т. е. цепочки с тремя сертификатами ниже корневого сертификата). Это означает, что между корневым сертификатом и листовым сертификатом могут существовать два «промежуточных сертификата». Такой компромисс реален, потому что сейчас каждая группа (см. R1) способна создавать произвольное число «промежуточных организаций» (см. рисунок С.1) и подписывать «промежуточные сертификаты» корневым закрытым ключом верхнего уровня. Каждая «промежуточная организация» может создать «сертификаты компании» для своих вторичных субъектов, что может быть использовано самим вторичным субъектом для создания листовых сертификатов (которые представляют собой, например, сертификаты контракта, выдаваемые потребителям).

D3 — Число корневых сертификатов: все существующие корневые сертификаты должны храниться на каждом транспортном средстве. Вследствие требования R3c предпочтительно небольшое число корневых сертификатов, но требование R4 предусматривает большое число корневых сертификатов. В качестве компромисса требование **[V2G2-008]** устанавливает, что необходим как минимум один корневой сертификат, но в примечании к нему рекомендуется минимум пять сертификатов. Это число (5) соответствует числу континентов. Должно быть возможным формирование вторичными субъектами каждого континента одной общей группы, учитывая, что это было достигнуто для EURO5/EU5. В этом случае для Европы был создан единый доверительный центр с единственным корневым сертификатом.

D4 — Действие корневых сертификатов: вследствие требования R1 (статические) корневые сертификаты (уже установленные при производстве транспортного средства) должны иметь почти неограниченный срок действия. Учитывая срок эксплуатации транспортного средства, **[V2G2-012]** требует наличия в любой момент времени корневого сертификата, действительного в течение не менее чем 35 лет. Для исключения слишком частой замены корневых сертификатов (управление сертификатами, см. R4 и R5) **[V2G2-011]** требует 40-летнего срока действия вновь созданного корневого сертификата.

D5 — Действие сервисных сертификатов изготовителя: сервисные сертификаты изготовителя также являются статическими сертификатами. По тем же причинам, которые изложены в D4, для новых сервисных сертификатов изготовителя требуется срок действия 30 лет.

D6 — Установка сертификатов контракта: как описано в требовании R1, оптимальным решением для изготовителя является установка всех сертификатов при производстве транспортного средства. Из-за R6 невозможно использовать сервисный сертификат изготовителя непосредственно для зарядки (на основе контракта), поскольку это исключает возможность работы EVSE в офлайновом режиме. Причина заключается в том, что без связи с серверной системой невозможно проверить наличие контракта для транспортного средства, которое передает только статический сервисный сертификат изготовителя (уже установленный при производстве транспортного средства независимо от существования контракта на зарядку).

a) В связи с этим сертификаты контракта должны быть установлены на транспортное средство и должны сменяться время от времени. Для исключения затрат для потребителя, изготовителя транспортного средства и вторичного субъекта это должно происходить автоматически через протокол зарядки. **[V2G2-235]** и **[V2G2-237]** требуют поддержки сообщений CertificateInstallationReq/Res SECC. **[V2G2-228]** и **[V2G2-231]** требуют поддержки сообщений CertificateUpdateReq/Response SECC. Указанные сообщения делают возможной установку первого сертификата контракта или установку последующих сертификатов контракта (если срок действия сертификата контракта, установленного на транспортном средстве, скоро истечет).

b) Во всех случаях, когда невозможно осуществить установку сертификата автоматически, трудоемкость ручной установки сертификата контракта должна минимизироваться. Как описано в требовании R2, это может быть достигнуто использованием стандартизованного формата файла при отправке сертификата контракта потребителю в виде файла. Формат файла определен в **[V2G2-648]**.

D7 — Действие сертификатов контракта: сертификаты контракта на транспортном средстве не должны заменяться слишком часто для исключения лишних циклов записи в память EVCC (см. требование R3). Для исключения чрезмерно частой смены сертификатов контракта (например, при каждом сеансе зарядки) **[V2G2-104]** требует, чтобы минимальный срок действия такого сертификата был равен четырем неделям (если срок контракта не короче). Для случаев, когда сертификат контракта отправляется потребителю и должен быть установлен вручную на станции техобслуживания (см. требование R2 и решение D6b), его срок действия должен быть гораздо длиннее для исключения неприемлемых усилий и затрат для потребителя. Тогда срок действия сертификата контракта должен быть равен продолжительности действия контракта.

D8 — Срок действия сертификатов SECC: механизмы аннуляции сертификатов SECC не устанавливаются. Вместо этого сертификаты SECC являются краткосрочными сертификатами.

С.1.5 Обзор итоговой структуры сертификатов

На рисунке С.1 показана структура сертификатов и соответствующие сроки действия.

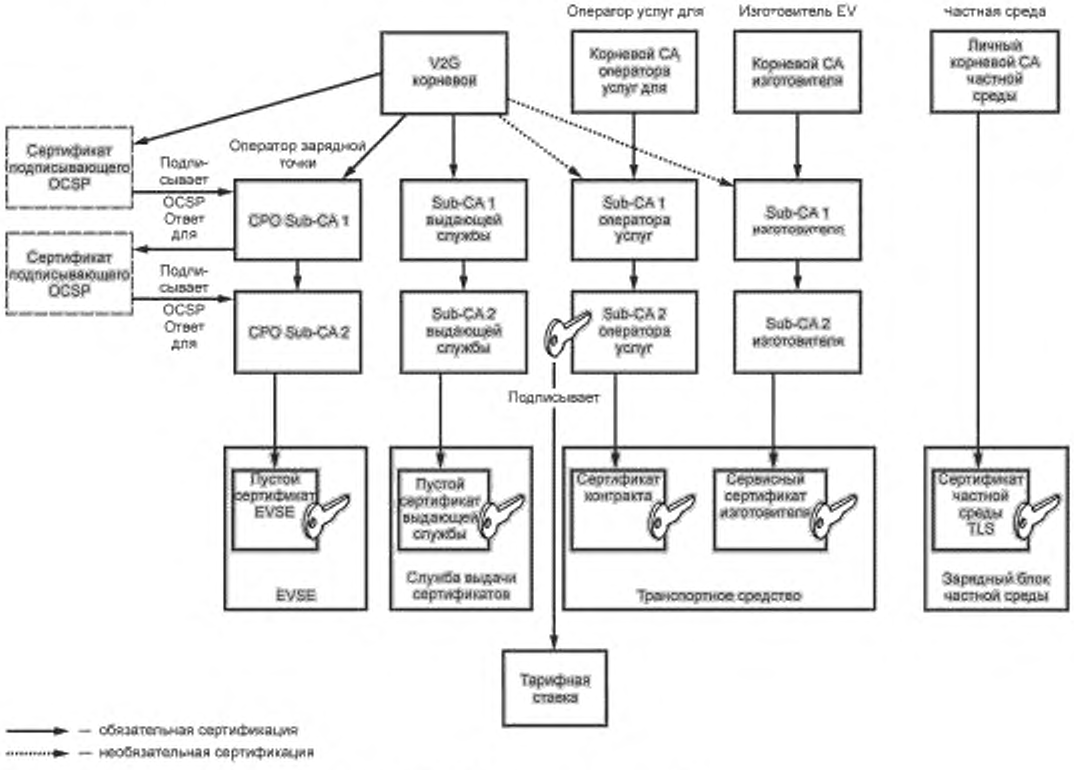


Рисунок С.1 — Обзор структуры сертификатов

На рисунке С.1 видно, что сервисные сертификаты изготовителя не зависят от PKI вторичных субъектов ниже корневых сертификатов. Корневой сертификат сервисного сертификата изготовителя создается самим изготовителем. Поэтому нет необходимости иметь (более длинную) цепочку сертификата. Разрешается повторно использовать корень V2G в качестве корневого сертификата оператора услуг для EV или корневого сертификата изготовителя (показано пунктирными линиями).

Все цепочки сертификатов имеют максимальную длину 3, т. е. речь идет о четырех сертификатах, включая корневой сертификат.

Цепочка сертификатов SECC передается EVCC для обеспечения проверки аутентичности SECC до установления соединения TLS (см. выше: исключение атак «посредника»).

Цепочка сертификата контракта передается в EVCC без корневого CA. Это ограничивает передачу тремя сертификатами, но это также означает, что транспортное средство не может проверить свой собственный сертификат контракта.

В С.2 дан пример упрощенного управления сертификатами в частной среде.

С.2 Упрощенное управление сертификатами в частной среде

С.2.1 Обзор (мотивация)

Из основных случаев использования настоящего стандарта вытекает, что EVSE является публичной зарядной станцией. Однако EVSE может также быть частным настенным блоком зарядки. Частный настенный блок зарядки находится в среде частной или малой компании (например, частный парковочный гараж, станция технического обслуживания или стоянка компании с собственным парком EV). Для обеспечения низкой стоимости производства и эксплуатации частного настенного блока зарядки необходимо учитывать его следующие особые характеристики:

- отсутствие соединения с серверной системой (за исключением этапа настройки во время установки);
- техническое обслуживание должно быть минимизированным, т. е. при нормальной эксплуатации вмешательства человека не требуется;

- никакой вторичный субъект не вовлечен. В частности, настенному блоку зарядки не требуется передавать какие-либо счета в серверную систему, потому что никакого специального учета энергии не выполняется.

Это означает, что OCSP и сертификаты с коротким сроком действия не могут быть здесь применены. Но даже в частной среде необходима коммуникация между SECC и EVCC, например для обеспечения передачи профилей зарядки (которые могут содержать данные с предпочтительным временем зарядки, т. е. ночью с 0:30 до 06:00, когда в наличии много энергии). Для обеспечения данного случая использования настоящий стандарт предусматривает специальные исключения.

С.2.2 Решение для частной среды

С.2.2.1 Общие положения

Для каждой локальной среды создается отдельный индивидуальный корневой СА частного оператора. Каждый настенный блок зарядки в данной локальной среде снабжается сертификатом SECC (подписанным корневым сертификатом частного оператора данной локальной среды).

Например, данные настенные блоки зарядки поставляются клиенту вместе с соответствующим корневым сертификатом частного оператора. Данный корневой сертификат частного оператора устанавливается на каждое транспортное средство, которое должно быть способно к зарядке в данной частной среде, например с использованием механизма сопряжения или других собственных механизмов изготовителя.



Рисунок С.2 — Зарядка в частной среде

EVCC хранит один корневой сертификат частного оператора для каждой частной среды, которую он хочет использовать. По сравнению с обычной структурой сертификатов V2G частная структура не позволяет наличия Sub-CA, и поэтому все листовые сертификаты непосредственно подписываются корневым сертификатом частного оператора.

Чтобы исключить частую смену сертификатов в настенных блоках зарядки, к их сертификатам SECC для TLS применяется специальная политика в отношении срока действия.

Срок действия сертификата SECC, установленного в частном настенном блоке зарядки и производного от корневого сертификата частного оператора, может быть таким, как определено частным оператором, в отличие от укороченных сроков действия сертификатов SECC.

Такие, почти произвольные, сроки действия не приводят к проблемам безопасности для других транспортных средств. Сертификат SECC принимается только транспортными средствами, которые имеют установленный соответствующий корневой сертификат частного оператора, т. е. транспортные средства данной частной среды. Таким образом, взломанный сертификат SECC вызывает проблемы только для транспортных средств, принадлежащих к данной индивидуальной частной среде. Оператор данной частной среды отвечает за предотвращение похищения сертификатов SECC настенных зарядных блоков, а также за принятие надлежащих мер в случае, если похищение сертификата все-таки произошло (см. С.2.2.4).

С.2.2.2 Установка частного корневого сертификата на транспортное средство

Оператор локальной среды обладает корневым сертификатом частного оператора, который принадлежит соответствующим настенным блокам зарядки (например, сертификат включен в объем поставки

настенных блоков зарядки). Данный корневой сертификат частного оператора должен быть установлен на все транспортные средства, которые должны иметь возможность заряжаться в данной частной среде. (При этом обеспечивается возможность проверки транспортным средством сертификата SECC настенного блока зарядки и выполнение им квитирования TLS.) Соответствующая процедура установки может быть выполнена разными способами.

Настоящий стандарт определяет простой механизм сопряжения: владелец сначала включает режим сопряжения EVCC, например нажатием кнопки или выбором соответствующей позиции меню. Данный режим действует в течение заданного времени. В то время как данный режим активен, транспортное средство подключается к EVSE и сертификат автоматически принимается и устанавливается в EVCC.

Установка сертификата может быть также осуществлена с использованием любого другого канала связи с транспортным средством, который изготовитель предлагает для этой цели. Эти каналы связи могут представлять собой онлайновый сервис изготовителя, диагностический интерфейс транспортного средства, используемый на станции технического обслуживания, интерфейс USB, имеющийся на транспортном средстве, и т. д.

С.2.2.3 Зарядка в частной среде

Для транспортного средства зарядка в частной среде совершенно такая же, как в среде, не являющейся частной. Это важно, поскольку транспортному средству не известно, где оно в данный момент находится. В частной среде настенный блок зарядки (частный) сертификат SECC передает EVCC, который проверяет (как всегда), действителен ли сертификат и является ли он производным от одного из корневых сертификатов, хранящихся на транспортном средстве. В таком порядке признаются корневые сертификаты V2G, а также корневые сертификаты частного оператора.

С.2.2.4 Взломанный сертификат настенного зарядного блока

Сертификат SECC частного настенного зарядного блока может быть взломан, например, если он похищен. Каждое транспортное средство, принадлежащее к данной частной среде (т. е. имеющее соответствующий корневой сертификат частного оператора), может подвергнуться атаке похищенным листовым сертификатом путем атаки «посредника» (безопасность всех транспортных средств, которые не принадлежат к данной частной среде, сохраняется). Такая атака возможна на каждом EVSE, даже вне частной среды. Поэтому в данном случае владелец частной среды обязан заблокировать корневой сертификат частного оператора на всех транспортных средствах частной среды. Для возобновления функции зарядки оператор должен сформировать совершенно новую структуру сертификатов, начиная с нового корневого сертификата частного оператора.

В данном сценарии важнейшими являются два аспекта:

a) сертификаты SECC настенных зарядных блоков должны быть надежно защищены или их похищение должно быть своевременно обнаружено;

b) усилия по замене корневых сертификатов частного оператора на всех транспортных средствах и всех настенных блоках зарядки данной среды в случае похищения должны быть приемлемыми.

Таким образом, описанное решение пригодно только для закрытых и ограниченных сред, т. е. небольшого числа транспортных средств и небольшого числа EVSE в частной среде или среде небольшого подразделения организации.

С.3 Использование сервисных сертификатов изготовителя

С.3.1 Введение

Трудность обращения с сертификатами контракта (в соответствии с определением в настоящем стандарте) изготовителя транспортных средств заключается в размещении таких сертификатов на транспортных средствах. Это становится необходимым во многих ситуациях, например при передаче транспортного средства потребителю, заключении контракта на энергию, смене оператора услуг для EV, замене во время ремонта транспортного средства компонента, содержащего сертификат контракта, истечении срока действия сертификатов контракта и т. д.

Использование диагностического прибора для записи файла, содержащего сертификат контракта (полученного потребителем от оператора услуг для EV при заключении контракта), на носитель транспортного средства нецелесообразно, поскольку такая ручная процедура на станции техобслуживания приводит к большим затратам. Другие решения, такие как установка через Web-страницу потребителя, также могут быть трудоемкими, связаны с возможными ошибками и требуют существования канала связи на борту транспортного средства.

Поэтому необходима автоматическая процедура установки такого сертификата контракта: выдача сертификата. Эта процедура поддерживается протоколом зарядки посредством сообщений CertificateInstallationReq/Res. Данные сообщения передают сертификат контракта от вторичного субъекта (например, оператора услуг для EV) на транспортное средство для установки и защищены за счет использования шифрованной коммуникации. Для обеспечения выдачи сертификатов дополнительно требуются действия, происходящие вне протокола зарядки. Эти действия показаны на рисунке С.3 и описываются следующим образом:

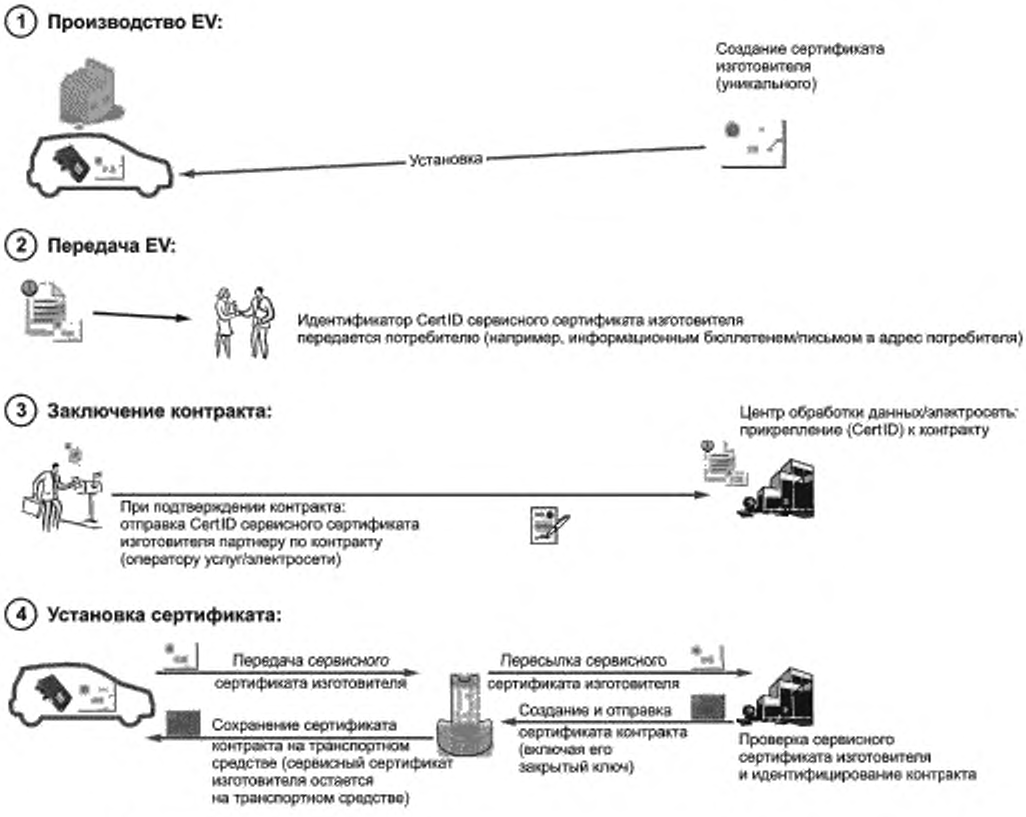


Рисунок С.3 — Действия, необходимые для предъявления сертификата изготовителя

С.3.2 Процессы

С.3.2.1 Производство EV

Во время производства EV уникальные сервисные сертификаты изготовителя и соответствующие закрытые ключи устанавливаются на каждом транспортном средстве (см. рисунок С.3, шаг 1). Каждый сервисный сертификат изготовителя содержит уникальный идентификатор (сокращенно «CertID») и является производным от сервисного корневого сертификата изготовителя, изданного изготовителем.

С.3.2.2 Передача EV

Во время передачи EV или перед ней потребителю сообщается его/ее CertID (см. рисунок С.3, шаг 2), например путем рассылки бюллетеней, помещения CertID в документацию EV, предложения онлайнового доступа к CertID и т. д.

С.3.2.3 Заключение контракта

При заключении контракта на энергию (см. рисунок С.3, шаг 3) потребитель передает CertID партнеру по контракту (оператору услуг для EV, сети и т. д.). Партнер по контракту присваивает (в своей IT-системе) контрактную информацию с использованием данного CertID. Кроме того, партнер по контракту создает сертификат контракта и присваивает сертификат контракта также с использованием данного CertID. Информация о существовании контракта для этого CertID передается в центр обработки данных (этой страны). Идеальный вариант — если вместе с сертификатом контракта (чтобы избежать задержек позднее, во время коммуникации на шаге 4).

С.3.2.4 Установка сертификата

При первой зарядке EV (или в любой ситуации, когда оно не имеет действительного сертификата контракта) оно передает свой сервисный сертификат изготовителя SECC с использованием сообщения CertificateInstallationReq (см. рисунок С.3, шаг 4). SECC пересылает это сообщение или сертификат в центр обработки данных (всем известным или собственным операторам услуг для EV, если в данной стране не существует центра обработки данных). Центр обработки данных (i) проверяет, является ли сервисный сертификат изготовителя аутентичным (т. е. действительным), с использованием корневого сертификата изготовителя и (ii) устанавливает подлинность сертификата, проверяет, зарегистрирован ли контракт для данного сервисного сертификата. Если соответствующий

сертификат контракта не был отправлен в центр обработки данных на шаге 3, центр обработки данных запрашивает сертификат контракта у оператора услуг для EV, заключившего контракт. Затем сертификат контракта (включая цепочку сертификата, необходимую для валидации) и соответствующий зашифрованный закрытый ключ отправляют через SECC на EV с помощью сообщения CertificateInstallationRes (см. ниже). Данная процедура выдачи сертификата используется во всех вышеуказанных случаях, например при заключении нового контракта, смене оператора услуг для EV и т. д., или рассматривается в ситуациях, если сертификат контракта на EV отсутствует, срок действия сертификата контракта истек или он был аннулирован, т. е. в любой ситуации, когда EV не имеет действительного сертификата контракта.

С.3.2.5 Обновление сертификата

Если срок действия текущего сертификата контракта еще не истек, он используется в сообщениях CertificateUpdateReq/Response для обновления данного сертификата контракта. Это также позволяет избежать ручной замены сертификата контракта с истекшим сроком действия. Сообщения для обновления сертификатов могут даже использоваться EV, которые совсем не имеют сервисного сертификата изготовителя или не реализуют процедуру выдачи. Однако если срок действия сертификата контракта истек (например, потому что EV в течение длительного времени не заряжалось на публичной/онлайновой зарядной станции), необходима вышеописанная процедура выдачи для обновления сертификата контракта (или ручная процедура).

С.3.2.6 Замена компонента

Когда при ремонте EV компонент, содержащий сертификат контракта, необходимо заменить, новый компонент может содержать иной сервисный сертификат изготовителя с другим CertID. Причина заключается в том, что закрытый ключ, принадлежавший изначальному сервисному сертификату изготовителя, хранился только внутри дефектного компонента и теперь утрачен. В таком случае потребитель (или станция технического обслуживания) сообщает о замене CertID всем заинтересованным сторонам. Затем текущий сертификат контракта может быть установлен с использованием стандартной процедуры установки сертификата (см. С.3.2.4). Это означает, что даже после ремонта EV ручная обработка сертификатов контракта не требуется.

С.4 Средства безопасности и соответствующие сертификаты

В данном пункте дается обзор разных случаев использования средств безопасности.

Таблица С.1 — Случаи использования средств безопасности и соответствующие сертификаты

Средства безопасности	Примечания	Удостоверения		
		EVCC (EV)	SECC (EVSE)	Вторичный субъект
Упрощенное управление сертификатами в частной среде См. С.2	Зарядка постоянным/переменным током с упрощенным управлением ключами - Аутентификация TLS SECC - Обмен конфиденциальными данными между EV и EVSE на основе TLS (обнаружение сервиса, передача энергии, учет) - Защита данных обмена между EV и EVSE на основе TLS (обнаружение сервиса, передача энергии, учет) TLS: AES (улучшенный стандарт шифрования) используется для шифрования потока/сеанса, протокол ECDH используется для обмена ключами	Закрытый корневой сертификат частной среды CA	Пара ключей TLS частной среды (закрытый ключ TLS частной среды и сертификат), удостоверенные закрытым корнем частной среды Закрытый корневой сертификат CA частной среды (только для передачи его на EV)	Пара ключей корневого CA частного оператора
Зарядка постоянным/переменным током с EIM (внешние средства идентификации) - Транспортная безопасность - Оплата на EVSE	- Аутентификация TLS SECC - Обмен конфиденциальными данными между EV и EVSE (обнаружение сервиса, передача энергии, учет) - Защита данных обмена между EV и EVSE на основе TLS (обнаружение сервиса, передача энергии, учет) TLS: AES (улучшенный стандарт шифрования) используется для шифрования потока/сеанса, протокол ECDH используется для обмена ключами	Корневые сертификаты V2G	Пара листовых ключей SECC (до 10 генерирований в зависимости от устаревания корневого ключа V2G) (листовой закрытый ключ и сертификат SECC для TLS), удостоверенные Sub-CA, который был удостоверен корневым CA V2G Все сертификаты Sub-CA OЗТ цепочки сертификатов ниже листового сертификата SECC Квитанция(и) ответа OCSP для соответствующего Sub-CA OЗТ	Нет

Продолжение таблицы С.1

Средства безопасности	Примечания	Удостоверения		
		EVCC (EV)	SECC (EVSE)	Вторичный субъект
Зарядка постоянным/переменным током с «Plug and Charge» - Зарядка постоянным/переменным током с применением средств обеспечения безопасности передачи - Оплата на основе контрактов с квитанцией прибора учета - Индивидуальные/динамические таблицы тарифов для потребителя	- Аутентификация TLS SECC - Аутентификация контракта на уровне приложения - Обмен конфиденциальными данными между EV и EVSE на основе TLS (обнаружение сервиса, передача энергии, учет) - Сохранность данных обмена между EV и EVSE на основе TLS (обнаружение сервиса, передача энергии, учет) - Подписание квитанции прибора учета на основе пары ключей контракта Уровень приложения: AES (улучшенный стандарт шифрования) для шифрования данных, протокол ECDH для обмена ключами, ECC для цифровой подписи для подписания тарифа TLS: AES (улучшенный стандарт шифрования) используется для шифрования потока/сеанса, протокол ECDH используется для обмена ключами	Корневые сертификаты V2G Пара ключей контракта (закрытый ключ контракта и сертификат для оплаты на основе контракта), удостоверенные CA, который может быть удостоверен корневым CA V2G Все сертификаты Sub-CA оператора услуг для EV цепочки сертификатов ниже сертификата контракта	Пара листовых ключей SECC (до 10 генерирований в зависимости от устаревания корневого ключа V2G) (закрытый ключ листового сертификата SECC и сертификат для TLS), удостоверенные Sub-CA, который был удостоверен корневым CA V2G Все сертификаты Sub-CA O3T цепочки сертификатов ниже листового сертификата SECC Все корневые сертификаты CA операторов услуг для EV, с которыми O3T имеет контракты Квитанция(и) ответа OCSP для соответствующего Sub-CA O3T Корневой сертификат V2G (для рассматриваемого региона)	Корневые сертификаты V2G Корневой сертификат CA оператора услуг для EV (если не ниже корня V2G) Все сертификаты Sub-CA операторов услуг для EV цепочки сертификатов ниже пары ключей Sub-CA оператора услуг для EV, выдающего сертификаты контракта Пара ключей Sub-CA оператора услуг для EV (закрытый ключ и сертификат SA) для подписания сертификата контракта
Зарядка постоянным/переменным током с PnC с динамическими тарифами, предоставлением и обновлением контрактных ключей - Зарядка постоянным/переменным током с применением средств обеспечения безопасности передачи - Оплата на основе контракта, с квитанцией прибора учета	- Аутентификация SECC на основе TLS - Обмен конфиденциальными данными между EV и EVSE (обнаружение сервиса, передача энергии, учет) - Сохранность данных и EVSE (обнаружение сервиса, передача энергии, учет) - Сохранность тарифных данных на основе цифровой подписи	Корневые сертификаты V2G Пара ключей контракта (закрытый ключ контракта и сертификат для оплаты на основе контракта), удостоверенные CA, который может быть удостоверен корневым CA V2G Все сертификаты Sub-CA оператора услуг для EV цепочки сертификатов ниже сертификата контракта	Пара листовых ключей SECC (до 10 генерирований в зависимости от устаревания корневого ключа V2G) (закрытый ключ листового сертификата SECC и сертификат для TLS), удостоверенные Sub-CA, который был удостоверен корневым CA V2G Все сертификаты Sub-CA O3T цепочки сертификатов ниже листового сертификата SECC	Корневые сертификаты V2G Корневой сертификат CA оператора услуг для EV (если не ниже корня V2G) Все сертификаты Sub-CA операторов услуг для EV цепочки сертификатов ниже пары ключей Sub-CA оператора услуг для EV, выдающего сертификаты контракта

Окончание таблицы С.1

Средства безопасности	Примечания	Удостоверения		
		EVCC (EV)	SECC (EVSE)	Вторичный субъект
- Индивидуальные/динамические таблицы тарифов для потребителя - Обновление и выдача сертификатов контракта	- Сохранность контрактных ключей/сертификатов на основе цифровой подписи - Конфиденциальность закрытого ключа новых сертификатов контракта на основе криптографии Прикладной уровень: AES (улучшенный стандарт шифрования) для шифрования данных, протокол ECDH используется для обмена ключами, ECC для цифровой подписи для подписания тарифа и выдачи контракта TLS: AES (улучшенный стандарт шифрования) для шифрования потока/сеанса, протокол ECDH используется для обмена ключами	Пара ключей сервисного сертификата изготовителя (закрытый ключ сервисного сертификата изготовителя и сертификат для выдачи контракта)	Все корневые сертификаты CA операторов услуг для EV, с которыми O3T имеет контракты Квитанция(и) ответа OCSP для соответствующего Sub-CA O3T Корневой сертификат V2G (для рассматриваемого региона)	Пара ключей Sub-CA оператора услуг для EV (закрытый ключ SA и сертификат) для подписания сертификата контракта и тарифа Пара ключей контракта, подлежащая передаче на EV Все сервисные сертификаты Sub-CA сертификата, завершающегося корнем V2G Пара сервисных ключей листового сертификата (листовой сервисный закрытый ключ и сертификат для установки/обновления) контракта, удостоверенные Sub-CA, который был удостоверен корневым CA V2G Сертификаты корневого CA изготовителя (если не ниже корня V2G) Все сертификаты Sub-CA изготовителя цепочки сертификатов ниже сервисного сертификата изготовителя Сервисный сертификат изготовителя для шифрования пары ключей контракта

Приложение D
(справочное)

Шифрование для распределения секретных ключей

D.1 Обзор

В соответствии с ISO 15118 сертификаты контракта создаются и затем передаются от SA к EVCC. Чтобы минимизировать двусторонний обмен в прямом и обратном направлениях, время обработки коммуникации, а также облегчить организационные процессы, было решено, что SA (а не EVCC) генерирует пару ключей для сертификата, затем сам соответствующий сертификат и в заключение посылает сертификат и закрытый ключ EVCC. Поскольку необходимо сохранять конфиденциальность закрытого ключа, должны использоваться надлежащие механизмы.

Для этого ISO 15118 использует протокол согласования эфемерно-статических ключей Диффи — Хеллмана для получения ключа сеанса в тех случаях, когда ключи получателя статические. SA затем использует этот ключ сеанса для передачи закрытого ключа в зашифрованной форме EVCC.

Поскольку ECDSA как вариант алгоритма цифровой подписи DSA на основе эллиптических кривых уже используется в ISO 15118, то такая же эллиптическая кривая, как для ECDSA, также используется для протокола эфемерно-статических ключей Диффи — Хеллмана. Кроме того, пара ключей, используемая EVCC для ECDSA, также используется как статическая пара ключей для протокола согласования ключа.

D.2 Согласование ключа по эфемерно-статическому протоколу Диффи — Хеллмана

Протокол согласования ключа Диффи — Хеллмана описывает, как между двумя субъектами может быть установлен общий секрет с использованием асимметричных криптографических алгоритмов по незащищенному каналу. Обе стороны затем могут вычислить общий секрет, из которого получается ключ сеанса, с помощью функции получения ключей. Для достижения вышеуказанных целей по двустороннему обмену и времени обработки коммуникации был выбран протокол эфемерно-статических ключей Диффи — Хеллмана.

В эфемерно-статическом варианте протокола Диффи — Хеллмана открытый ключ получателя (EVCC) не меняется, т. е. является статическим, и уже известен отправителю (SA). Отправитель, однако, по-прежнему использует эфемерный открытый ключ, который передается получателю. Таким образом, как отправитель, так и получатель могут получить один и тот же ключ сеанса без ответного сообщения от получателя. Требуется только одно коммуникационное сообщение (см. рисунок D.1).

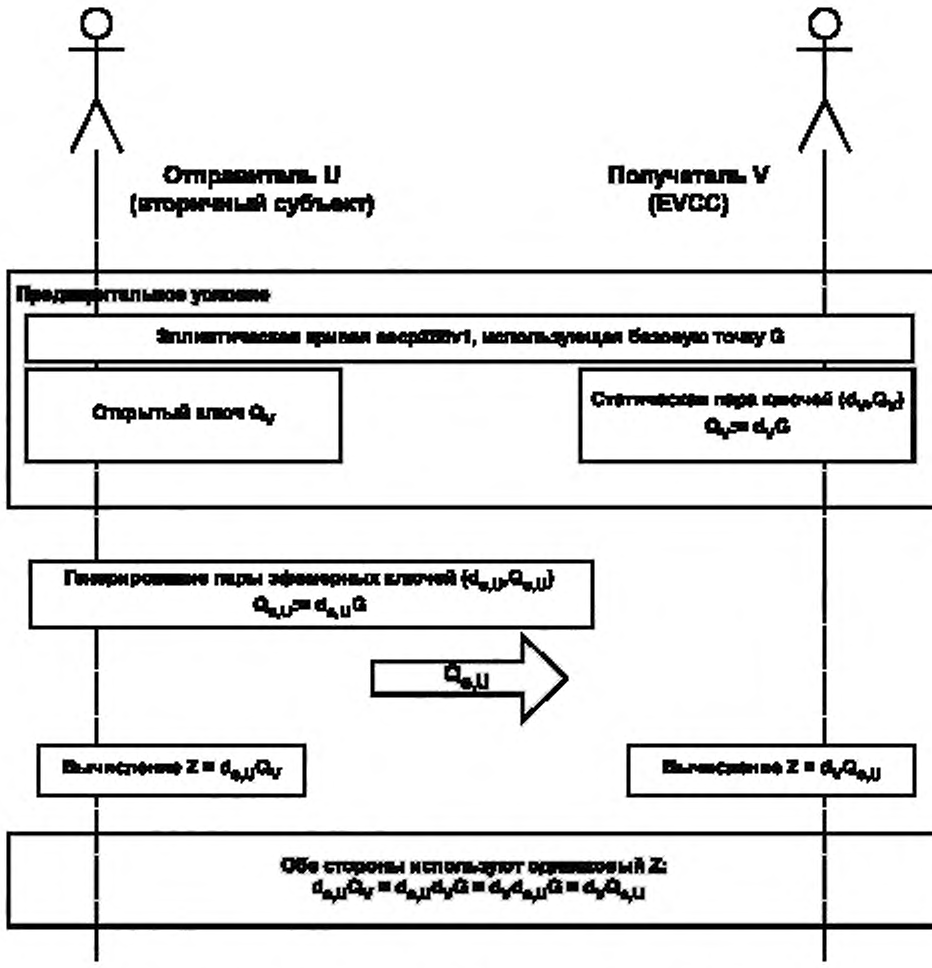


Рисунок D.1 — Согласование ключа по эфемерно-статическому протоколу Диффи — Хеллмана (информационное изображение, не использовать для реализации)

D.3 Пары ключей

EVCC повторно использует пару ключей, которую он также использует для подписей ECDSA, т. е. пару ключей, чей открытый ключ хранится в его текущем сертификате контракта или в сервисном сертификате изготовителя (в зависимости от типа сообщения). В обоих случаях сертификат и его открытый ключ уже известны вторичному субъекту. SA может, таким образом, предварительно рассчитать новые сертификаты и закрытые ключи для EVCC и зашифровать их ключом сеанса без какого-либо сообщения от EVCC. Когда EVCC соединяется со вторичным субъектом (через SECC), предварительно рассчитанное шифрованное сообщение может быть извлечено и дешифровано после только одного сообщения от SA, без какой-либо криптографической обработки со стороны SA и без какого-либо дополнительного сообщения от EVCC.

SA использует эфемерные ключи, и поэтому ключи сеанса будут разными для каждой отправки сертификата и закрытого ключа. Следует учесть, что протокол эфемерно-статических ключей Диффи — Хеллмана не обеспечивает совершенную секретность передачи данных, т. е. если статический закрытый ключ EVCC перехвачен злоумышленником, то этот злоумышленник может дешифровать имевшую место в прошлом, а также будущую коммуникацию между SA и EVCC, а еще и генерировать подписи, поскольку ключ для подписей такой же, как ключ для шифрования.

Приложение Е
(справочное)

Обзор XML-подписей

Е.1 Обзор

Согласно ISO 15118 XML-подписи используются для подписания, например, тарифных ставок, показаний приборов учета или целых тел сообщений. Для понимания того, как XML-подписи используются в ISO 15118, приведен подробный пример с использованием действительных данных, ключей и значений подписи.

Е.2 Генерирование подписей

Подписание данных с использованием XML-подписей требует двух основных шагов: генерирование элемента Reference и генерирование подписи. Генерирование элемента Reference включает все шаги, необходимые для добавления неких XML-данных, которые должны быть подписаны, к XML-элементу SignedInfo. Генерирование подписи затем вычисляет значение подписи.

Все значения EXI, хэш-кода и подписей, которые даны в этом примере, действительны и могут быть проверены с использованием следующей пары ключей:

- открытый ключ (исходные данные x, y):

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	43	E4	FC	4C	CB	64	39	04	27	9C	7A	5E	65	76	B3	23
00000010	E5	5E	C7	9F	F0	E5	A4	05	6E	33	40	84	CB	C3	36	FF
00000020	46	E4	4C	1A	DD	F6	91	62	E5	19	2C	2A	83	FC	2B	CA
00000030	9D	8F	46	EC	F4	B7	80	67	C2	47	6F	6B	3F	34	60	0E

- закрытый ключ (исходные данные x):

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	B9	13	49	63	F5	1C	44	14	73	84	35	05	7F	97	BE	F1
00000010	01	0C	AB	CB	8D	BD	E9	C5	D4	81	38	39	6A	A9	4E	9D

Следующий пример показывает полный процесс, необходимый для генерирования действительной подписи для сообщения AuthorizationReq со следующим содержанием:

```
<v2gci_tc:AuthorizationReq v2gci_tc:Id="ID1">
    <v2gci_tc:GenChallenge>U29tZSBSYWNkb20gRGF0YQ==</v2gci_tc:GenChallenge>
</v2gci_tc:AuthorizationReq>
```

Следующий XML-фрагмент может использоваться в качестве базового шаблона для ISO 15118 при генерировании XML-подписей. Необходимые идентификаторы алгоритма и минимально необходимые элементы уже добавлены. Для действительной подписи требуется только расширение элемента <Reference> и конкретной свертки и значений подписи.

```
<xmlsig:Signature>
    <xmlsig:SignedInfo>
        <xmlsig:CanonicalizationMethod Algorithm="http://www.w3.org/TR/canonical-exi/"/>
        <xmlsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256"/>
        <xmlsig:Reference URI="#ID1">
                <xmlsig:Transforms>
                    <xmlsig:Transform Algorithm="http://www.w3.org/TR/canonical-exi/"/>
                </xmlsig:Transforms>
            <xmlsig:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <xmlsig:DigestValue></xmlsig:DigestValue>
        </xmlsig:Reference>
    </xmlsig:SignedInfo>
    <xmlsig:SignatureValue></xmlsig:SignatureValue>
</xmlsig:Signature>
```

Первым шагом для генерирования подписи является добавление XML-элемента Reference к данным, которые должны быть подписаны. Для этого атрибут URI устанавливается в значение идентификатора элемента и данные трансформируются по правилам EXI (трансформация элемента Reference) с использованием фрагментов грамматики в соответствии со схемой EXI, в основе которой лежит схема V2G_CI_MsgDef.

Получаемый поток EXI для данного AuthorizationReq:

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   80 04 01 52 51 0C 40 82 9B 7B 6B 29 02 93 0B 73
00000010   23 7B 69 02 23 0B A3 09 E8
```

Следующим шагом является хэширование EXI-потока с использованием безопасного алгоритма хэширования SHA256. Это значение хэш-кода затем кодируется по методу base64 и вместе с URI добавляется к элементу SignedInfos Reference в качестве DigestValue (выделено красным цветом).

```
<xmlsig:Signature>
    <xmlsig:SignedInfo>
        <xmlsig:CanonicalizationMethod Algorithm="http://www.w3.org/TR/canonical-exi/"/>
        <xmlsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256"/>
        <xmlsig:Reference URI="#ID1">
            <xmlsig:Transforms>
              <xmlsig:Transform Algorithm="http://www.w3.org/TR/canonical-exi/"/>
            </xmlsig:Transforms>
            <xmlsig:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <xmlsig:DigestValue>0bXgPQBvuMMDmERTBR81TKGPwOCRYXT4e8d8mPSqk=</xmlsig:DigestValue>
        </xmlsig:Reference>
    </xmlsig:SignedInfo>
    <xmlsig:SignatureValue></xmlsig:SignatureValue>
</xmlsig:Signature>
```

Примечание 1 — При необходимости этот шаг повторяется для всех идентификаторов (references), которые должны быть добавлены к подписи (например, для нескольких тарифных ставок).

Примечание 2 — В отличие от текстового XML в представлении данного примера EXI кодирует двоичные данные собственным кодом, поэтому base64-кодирование там не применяется.

После генерирования элемента Reference следующим шагом является генерирование подписи. Для этого шага исходные данные уже не требуются. XML-подписью подписывается только элемент SignedInfo. Поэтому этот элемент должен быть закодирован по правилам EXI с использованием информированной о схеме грамматики фрагмента на основе XMLdsig-схемы (канонизация). Получаемый EXI-поток следующий:

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   80 81 12 B4 3A 3A 38 1D 17 97 BB BB BB 97 3B 99
00000010   97 37 B9 33 97 AA 29 17 B1 B0 B7 37 B7 34 B1 B0
00000020   B6 16 B2 BC 34 97 A1 AB 43 A3 A3 81 D1 79 7B BB
00000030   BB B9 73 B9 99 73 7B 93 39 79 91 81 81 89 79 81
00000040   A1 7B C3 6B 63 23 9B 4B 39 6B 6B 7B 93 29 1B 2B
00000050   1B 23 9B 09 6B 9B 43 09 91 A9 B2 20 62 34 94 43
00000060   10 25 68 74 74 70 3A 2F 2F 77 77 77 2E 77 33 2E
00000070   6F 72 67 2F 54 52 2F 63 61 6E 6F 6E 69 63 61 6C
00000080   2D 65 78 69 2F 48 52 D0 E8 E8 E0 74 5E 5E EE EE
00000090   EE 5C EE 66 5C DE E4 CE 5E 64 60 60 62 5E 60 68
000000A0   5E F0 DA D8 CA DC C6 46 E6 D0 C2 64 6A 6C 84 1A
000000B0   36 BC 07 A0 0C B7 DC AD 66 2F 30 88 A6 0A 3D 6A
000000C0   99 43 1F 81 C1 22 C2 E9 F1 67 8E F5 31 E9 55 23
000000D0   70
```

Этот поток затем хэшируется с использованием SHA256. Получаемое значение хэш-кода:

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   A4 E9 03 E1 82 43 04 1B 55 4E 11 64 7E 10 1E D2
00000010   5F C9 F2 15 2A F4 67 40 14 FE 2A DE AC 1E 1C F7
```

Это значение хэш-кода используется как входные данные для алгоритма подписи (ECDSA с кривой secp256r1). Вычисление подписи выполняется с помощью закрытого ключа. Полученное значение подписи затем кодируется по методу base64 и добавляется как SignatureValue к XML-данным подписи (выделены красным цветом):

```
<xmlsig:Signature>
    <xmlsig:SignedInfo>
        <xmlsig:CanonicalizationMethod Algorithm="http://www.w3.org/TR/canonical-exi/"/>
        <xmlsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256"/>
        <xmlsig:Reference URI="#ID1">
            <xmlsig:Transforms>
                <xmlsig:Transform Algorithm="http://www.w3.org/TR/canonical-exi/"/>
            </xmlsig:Transforms>
            <xmlsig:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <xmlsig:DigestValue>0bXgPQBlvuVrMXmERTBR61TKGPwOCRYXT4s8d6mPSqk=</xmlsig:DigestValue>
        </xmlsig:Reference>
    </xmlsig:SignedInfo>
    <xmlsig:SignatureValue>TI8gwUALpnYGqkgRVyovGtPBUInZVCA2NDC7JrSdsQTwjfqL+AVeY6S3Wo
                          0xaSBvqNVDCLpY8FZrlrr2ks5ZUA==</xmlsig:SignatureValue>
</xmlsig:Signature>
```

Объединение всего указанного вместе дает следующее сообщение V2G, которое будет передано от EVCC к SECC:

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmlsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9988AABBCCDDEEFF</v2gci_h:SessionID>
        <xmlsig:Signature>
            <xmlsig:SignedInfo>
                <xmlsig:CanonicalizationMethod Algorithm="http://www.w3.org/TR/canonical-exi/"/>
                <xmlsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256"/>
                <xmlsig:Reference URI="#ID1">
```

```
                    <xmlsig:Transforms>
                        <xmlsig:Transform Algorithm="http://gost.e-ecolog.ru                    xi/"/>
                    </xmlsig:Transforms>
                    <xmlsig:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
                    <xmlsig:DigestValue>0bXgPQBlvuVrMXmERTBR61TKGPwO
                                        CRYXT4s8d6mPSqk=</xmlsig:DigestValue>
                </xmlsig:Reference>
            </xmlsig:SignedInfo>
            <xmlsig:SignatureValue>TI8gwUALpnYGqkgRVyovGtPBUInZVCA2NDC7JrSdsQTwjfqL+AVeY6S3
                                   Wo0xaSBvqNVDCLpY8FZrlrr2ks5ZUA==</xmlsig:SignatureValue>
        </xmlsig:Signature>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:AuthorizationReq v2gci_b:Id="ID1">
            <v2gci_b:GenChallenge>U29tZSBSYW5kb20gRGF0YQ==</v2gci_b:GenChallenge>
        </v2gci_b:AuthorizationReq>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 24 сообщения V2G — Генерирование подписи для сообщения AuthorizationReq

Е.3 Генерирование подписей для вторичных субъектов

Для тарифных ставок генерирование подписей должно выполняться вторичным субъектом. Поэтому некоторую информацию о сеансе зарядки требуется передать от SECC вторичному субъекту до генерирования заключительного сообщения ChargeParameterDiscoveryRes. Эта информация обязательно включает используемую версию схемы, необходимое количество энергии, время убытия, перечень корневых сертификатов EV и, если требуется, некоторые дополнительные данные EVSE, которые могут быть использованы для генерирования тарифных ставок. SA должен получить XSD-файлы для используемой версии схемы. Во время генерирования подписи данные файлы могут использоваться вместе с базовым EXI-генератором, например EXIficient, для создания потоков грамматики фрагментов EXI во время трансформации элементов <Reference> и канонизации <SignedInfo>.

Тарифные ставки создаются как XML-данные, которые действительны в соответствии с используемой версией схемы. Обособленная XML-подпись генерируется вторичным субъектом (см. [V2G2-307]), как указано в последовательности операций выше. Используемый сертификат должен быть производным от корневого сертификата EV, и данная ключевая информация может быть также добавлена к подписи.

XML-данные (тарифные ставки и подпись) передаются SECC произвольным способом. ChargeParameterDiscoveryRes, которое включает полученные тарифные ставки и соответствующую подпись, затем генерируется SECC, кодируется по правилам EXI и передается EVCC (см. [V2G2-308] и [V2G2-309]).

Примечание — Одно значение подписи используется для подписания всех тарифных ставок. В этом сценарии несколько элементов <Reference> включаются в элемент <SignedInfo>.

Е.4 Валидация подписей

Валидация XML-подписей требует также двух основных шагов: валидации элемента Reference и валидации подписи. Во время валидации элемента Reference сначала вычисляется канонизация элемента SignedInfo. Это означает, что элемент SignedInfo кодируется по правилам EXI с использованием грамматики фрагмента на основе XMLdsig-схемы.

Затем проверяется элемент DigestValue для каждого элемента Reference. Это требует разыменования URI в качестве первого шага. Исходные подписанные данные необходимо вернуть и трансформировать с использованием грамматики фрагмента на основе схемы V2G_CI_MsgDef. Сгенерированный EXI-поток затем хэшируется с использованием алгоритма SHA256 и сравнивается с полученным элементом DigestValue.

Если валидация элемента Reference проходит для всех элементов Reference, значение подписи валидируется. Канонизация/EXI-поток, который представляет элемент SignedInfo, хэшируется с использованием алгоритма SHA256, и на основе этого значения полученное SignatureValue валидируется с использованием открытого ключа.

Приложение F
(рекомендуемое)

Определения схем

F.1 Обзор

Спецификация сообщений на уровне приложения V2G состоит из документов XML-схемы:
- «V2G_CI_AppProtocol»: определяет сообщения подтверждения установления связи по протоколу;
- «V2G_CI_MsgDef»: определяет структуру сообщений;
- «V2G_CI_MsgHeader»: определяет заголовок сообщений;
- «V2G_CI_MsgBody»: определяет тело сообщений;
- «V2G_CI_MsgDataTypes»: определяет типы данных;
- «xmldsig-core-schema»: определяет схему W3C для XML-подписей.

На рисунке С.1 дана диаграмма зависимости для документов XML-схемы.

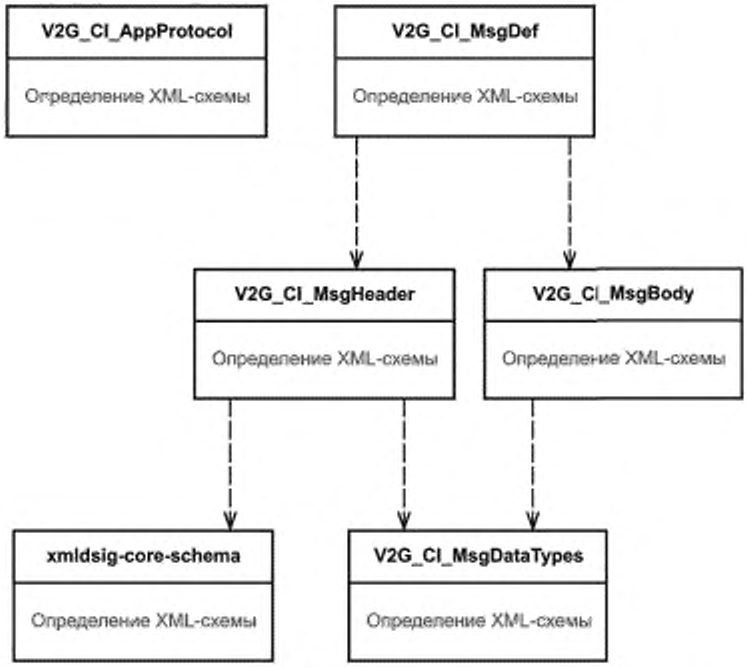


Рисунок F.1 — Диаграмма зависимости определений XML-схемы V2G

F.2 V2G_CI_AppProtocol.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="urn:iso:15118:2:2010:AppProtocol"
    targetNamespace="urn:iso:15118:2:2010:AppProtocol">
    <xs:element name="supportedAppProtocolReq">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="AppProtocol" type="AppProtocolType" maxOccurs="20"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="supportedAppProtocolRes">
        <xs:complexType>
```

```
            <xs:sequence>
                <xs:element name="ResponseCode" type="responseCodeType"/>
                <xs:element name="SchemaID" type="idType" minOccurs="0"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:complexType name="AppProtocolType">
        <xs:sequence>
            <xs:element name="ProtocolNamespace" type="protocolNamespaceType"/>
            <xs:element name="VersionNumberMajor" type="xs:unsignedInt"/>
            <xs:element name="VersionNumberMinor" type="xs:unsignedInt"/>
            <xs:element name="SchemaID" type="idType"/>
            <xs:element name="Priority" type="priorityType"/>
        </xs:sequence>
    </xs:complexType>
    <xs:simpleType name="idType">
        <xs:restriction base="xs:unsignedByte"/>
    </xs:simpleType>
    <xs:simpleType name="protocolNameType">
        <xs:restriction base="xs:string">
            <xs:maxLength value="30"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="protocolNamespaceType">
        <xs:restriction base="xs:anyURI">
            <xs:maxLength value="100"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="priorityType">
        <xs:restriction base="xs:unsignedByte">
            <xs:minInclusive value="1"/>
            <xs:maxInclusive value="20"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="responseCodeType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="OK_SuccessfulNegotiation"/>
            <xs:enumeration value="OK_SuccessfulNegotiationWithMinorDeviation"/>
            <xs:enumeration value="Failed_NoNegotiation"/>
        </xs:restriction>
    </xs:simpleType>
</xs:schema>
```

F.3 V2G_CI_MsgDef.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:"urn:iso:15118:2:2013:MsgDef" xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody" targetNamespace="urn:iso:15118:2:2013:MsgDef" elementFormDefault
="qualified" attributeFormDefault="qualified" version="15118 2.0">
    <!-- attributeFormDefault="unqualified" -->
    <xs:import namespace="urn:iso:15118:2:2013:MsgHeader" schemaLocation="V2G_CI_MsgHeader.xsd"/>
    <xs:import namespace="urn:iso:15118:2:2013:MsgBody" schemaLocation="V2G_CI_MsgBody.xsd"/>
    <!-- Message Structure -->
    <xs:element name="V2G_Message">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Header" type="v2gci_h:MessageHeaderType"/>
                <xs:element name="Body" type="v2gci_b:BodyType"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

F.4 V2G_CI_MsgHeader.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="urn:iec:15118:2:2013:MsgHeader"
xmlns:v2gci_h="urn:iec:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="urn:iec:15118:2:2013:MsgHeader"
elementFormDefault="qualified" attributeFormDefault="qualified">
<xs:import namespace="urn:iec:15118:2:2013:MsgDataTypes" schemaLocation="V2G_CI_MsgDataTypes.xsd"/>
<xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="xmldsig-core-schema.xsd"/>
<!-- Message Header -->
    <xs:complexType name="MessageHeaderType">
        <xs:sequence>
            <xs:element name="SessionID" type="v2gci_t:sessionIDType"/>
            <xs:element name="Notification" type="v2gci_t:NotificationType" minOccurs="0"/>
            <xs:element ref="xmldsig:Signature" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

F.5 V2G_CI_MsgBody.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="urn:iec:15118:2:2013:MsgBody"
xmlns:v2gci_d="urn:iec:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iec:15118:2:2013:MsgDataTypes"
targetNamespace="urn:iec:15118:2:2013:MsgBody"
elementFormDefault="qualified" attributeFormDefault="qualified">
    <xs:import namespace="urn:iec:15118:2:2013:MsgDataTypes" schemaLocation="V2G_CI_MsgDataTypes.xsd"/>
    <!-- Body -->
    <xs:complexType name="BodyType">
        <xs:sequence>
            <xs:element ref="BodyElement" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="BodyElement" type="BodyBaseType" abstract="true"/>
    <xs:complexType name="BodyBaseType" abstract="true"/>
    <!-- ..... -->
    <!-- Common Messages (AC/DC) -->
    <!-- ..... -->
    <!--          -->
    <!-- Session Setup -->
    <!--          -->
    <xs:element name="SessionSetupReq" type="SessionSetupReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="SessionSetupReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="EVCCID" type="v2gci_t:evccIDType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="SessionSetupRes" type="SessionSetupResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="SessionSetupResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="EVSEID" type="v2gci_t:evseIDType"/>
                    <xs:element name="EVSETimeStamp" type="xs:long" minOccurs="0"/>
                </xs:sequence>
```

```
            </xs:extension>
        </xs:complexContent>
</xs:complexType>
<!--                -->
<!-- Service Discovery -->
<!--                -->
<xs:element name="ServiceDiscoveryReq" type="ServiceDiscoveryReqType" substitutionGroup="BodyElement"/>
<xs:complexType name="ServiceDiscoveryReqType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ServiceScope" type="v2gci_t:serviceScopeType" minOccurs="0"/>
                <xs:element name="ServiceCategory" type="v2gci_t:serviceCategoryType" minOccurs="0"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="ServiceDiscoveryRes" type="ServiceDiscoveryResType" substitutionGroup="BodyElement"/>
<xs:complexType name="ServiceDiscoveryResType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                <xs:element name="PaymentOptionList" type="v2gci_t:PaymentOptionListType"/>
                <xs:element name="ChargeService" type="v2gci_t:ChargeServiceType"/>
                <!--<xs:element name="ServiceList" type="v2gci_t:ServiceTagListType" minOccurs="0"/>-->
                <xs:element name="ServiceList" type="v2gci_t:ServiceListType" minOccurs="0"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!--                -->
<!-- Service Detail -->
<!--                -->
<xs:element name="ServiceDetailReq" type="ServiceDetailReqType" substitutionGroup="BodyElement"/>
<xs:complexType name="ServiceDetailReqType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ServiceID" type="v2gci_t:serviceIDType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="ServiceDetailRes" type="ServiceDetailResType" substitutionGroup="BodyElement"/>
<xs:complexType name="ServiceDetailResType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                <xs:element name="ServiceID" type="v2gci_t:serviceIDType"/>
                <xs:element name="ServiceParameterList" type="v2gci_t:ServiceParameterListType"
                            minOccurs="0"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!--                    -->
<!-- Service Payment & Selection -->
<!--                    -->
```

```
    <xs:element name="PaymentServiceSelectionReq" type="PaymentServiceSelectionReqType" substitutionGroup="
BodyElement"/>
    <xs:complexType name="PaymentServiceSelectionReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="SelectedPaymentOption" type="v2gci_t:paymentOptionType"/>
                    <xs:element name="SelectedServiceList" type="v2gci_t:SelectedServiceListType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="PaymentServiceSelectionRes" type="PaymentServiceSelectionResType" substitutionGroup="
BodyElement"/>
    <xs:complexType name="PaymentServiceSelectionResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--            -->
    <!-- Payment Details -->
    <!--            -->
    <xs:element name="PaymentDetailsReq" type="PaymentDetailsReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="PaymentDetailsReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="eMAID" type="v2gci_t:eMAIDType"/>
                    <xs:element name="ContractSignatureCertChain" type="v2gci_t:CertificateChainType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="PaymentDetailsRes" type="PaymentDetailsResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="PaymentDetailsResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="GenChallenge" type="v2gci_t:genChallengeType"/>
                    <xs:element name="EVSETimeStamp" type="xs:long"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                -->
    <!-- Authorization-->
    <!--                -->
    <xs:element name="AuthorizationReq" type="AuthorizationReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="AuthorizationReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="GenChallenge" type="v2gci_t:genChallengeType" minOccurs="0"/>
                </xs:sequence>
                <xs:attribute name="Id" type="xs:ID"/>
            </xs:extension>
        </xs:complexContent>
```

```
    </xs:complexType>
    <xs:element name="AuthorizationRes" type="AuthorizationResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="AuthorizationResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="EVSEProcessing" type="v2gci_t:EVSEProcessingType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                    -->
    <!-- Charge Parameter Discovery -->
    <!--                    -->
    <xs:element name="ChargeParameterDiscoveryReq" type="ChargeParameterDiscoveryReqType"
substitutionGroup="BodyElement"/>
    <xs:complexType name="ChargeParameterDiscoveryReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="MaxEntriesSASchedule Tuple" type="xs:unsignedShort" minOccurs="0"/>
                    <!-- new -->
                    <xs:element name="RequestedEnergyTransferMode" type="v2gci_t:EnergyTransferModeType"/>
                    <xs:element ref="v2gci_t:EVChargeParameter"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="ChargeParameterDiscoveryRes" type="ChargeParameterDiscoveryResType"
substitutionGroup="BodyElement"/>
    <xs:complexType name="ChargeParameterDiscoveryResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="EVSEProcessing" type="v2gci_t:EVSEProcessingType"/>
                    <xs:element ref="v2gci_t:SASchedules" minOccurs="0"/>
                    <xs:element ref="v2gci_t:EVSEChargeParameter"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                    -->
    <!-- Power Delivery     -->
    <!--                    -->
    <xs:element name="PowerDeliveryReq" type="PowerDeliveryReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="PowerDeliveryReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ChargeProgress" type="v2gci_t:chargeProgressType"/>
                    <xs:element name="SAScheduleTupleID" type="v2gci_t:SAIDType"/>
                    <xs:element name="ChargingProfile" type="v2gci_t:ChargingProfileType" minOccurs="0"/>
                    <xs:element ref="v2gci_t:EVPowerDeliveryParameter" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="PowerDeliveryRes" type="PowerDeliveryResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="PowerDeliveryResType">
```

```
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element ref="v2gci_t:EVSEStatus"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                    -->
    <!-- Metering Receipt   -->
    <!--                    -->
    <xs:element name="MeteringReceiptReq" type="MeteringReceiptReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="MeteringReceiptReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="SessionID" type="v2gci_t:sessionIDType"/>
                    <xs:element name="SASchedule TupleID" type="v2gci_t:SAIDType" minOccurs="0"/>
                    <xs:element name="MeterInfo" type="v2gci_t:MeterInfoType"/>
                </xs:sequence>
                <xs:attribute name="Id" type="xs:ID"/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="MeteringReceiptRes" type="MeteringReceiptResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="MeteringReceiptResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element ref="v2gci_t:EVSEStatus"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                -->
    <!-- SessionStop    -->
    <!--                -->
    <xs:element name="SessionStopReq" type="SessionStopReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="SessionStopReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ChargingSession" type="v2gci_t:chargingSessionType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="SessionStopRes" type="SessionStopResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="SessionStopResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                    -->
    <!-- Certificate Update -->
```

```
    <!--                    -->
    <xs:element name="CertificateUpdateReq" type="CertificateUpdateReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="CertificateUpdateReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ContractSignatureCertChain" type="v2gci_t:CertificateChainType"/>
                    <xs:element name="eMAID" type="v2gci_t:eMAIDType"/>
                    <xs:element name="ListOfRootCertificateIDs" type="v2gci_t1:ListOfRootCertificateIDsType"/>
                </xs:sequence>
                <xs:attribute name="Id" type="xs:ID" use="required"/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="CertificateUpdateRes" type="CertificateUpdateResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="CertificateUpdateResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="SAProvisioningCertificateChain" type="v2gci_t:CertificateChainType"/>
                    <xs:element name="ContractSignatureCertChain" type="v2gci_t:CertificateChainType"/>
                    <xs:element name="ContractSignatureEncryptedPrivateKey"
type="v2gci_t:ContractSignatureEncryptedPrivateKeyType"/>
                    <xs:element name="DHpublickey" type="v2gci_t:DiffieHellmanPublickeyType"/>
                    <xs:element name="eMAID" type="v2gci_t:EMAIDType"/>
                    <xs:element name="RetryCounter" type="xs:short" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                    -->
    <!-- Certificate Installation-->
    <!--                    -->
    <xs:element name="CertificateInstallationReq" type="CertificateInstallationReqType"
    substitutionGroup="BodyElement"/>
    <xs:complexType name="CertificateInstallationReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="OEMProvisioningCert" type="v2gci_t:certificateType"/>
                    <xs:element name="ListOfRootCertificateIDs" type="v2gci_t1:ListOfRootCertificateIDsType"/>
                </xs:sequence>
                <xs:attribute name="Id" type="xs:ID" use="required"/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="CertificateInstallationRes" type="CertificateInstallationResType"
    substitutionGroup="BodyElement"/>
    <xs:complexType name="CertificateInstallationResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="SAProvisioningCertificateChain" type="v2gci_t:CertificateChainType"/>
                    <!-- new -->
                    <xs:element name="ContractSignatureCertChain" type="v2gci_t:CertificateChainType"/>
                    <xs:element name="ContractSignatureEncryptedPrivateKey"
    type="v2gci_t:ContractSignatureEncryptedPrivateKeyType"/>
                    <xs:element name="DHpublickey" type="v2gci_t:DiffieHellmanPublickeyType"/>
                    <xs:element name="eMAID" type="v2gci_t:EMAIDType"/>
```

```
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!-- ..... -->
    <!-- AC-Messages -->
    <!-- ..... -->
    <!-- -->
    <!-- Charging Status -->
    <!-- -->
    <xs:element name="ChargingStatusReq" type="ChargingStatusReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="ChargingStatusReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence/>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="ChargingStatusRes" type="ChargingStatusResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="ChargingStatusResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="EVSEID" type="v2gci_t:evseIDType"/>
                    <xs:element name="SAScheduleTupleID" type="v2gci_t:SAIDType"/>
                    <xs:element name="EVSEMaxCurrent" type="v2gci_t:PhysicalValueType" minOccurs="0"/>
                    <xs:element name="MeterInfo" type="v2gci_t:MeterInfoType" minOccurs="0"/>
                    <xs:element name="ReceiptRequired" type="xs:boolean" minOccurs="0"/>
                    <xs:element name="AC_EVSEStatus" type="v2gci_t:AC_EVSEStatusType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!-- ..... -->
    <!-- DC-Messages -->
    <!-- ..... -->
    <!-- -->
    <!-- Cable Check -->
    <!-- -->
    <xs:element name="CableCheckReq" type="CableCheckReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="CableCheckReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="DC_EVStatus" type="v2gci_t:DC_EVStatusType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="CableCheckRes" type="CableCheckResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="CableCheckResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="DC_EVSEStatus" type="v2gci_t:DC_EVSEStatusType"/>
                    <xs:element name="EVSEProcessing" type="v2gci_t:EVSEProcessingType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
```

```
</xs:complexType>
<!--                    -->
<!-- Pre-Charge -->
<!--                    -->
<xs:element name="PreChargeReq" type="PreChargeReqType" substitutionGroup="BodyElement"/>
<xs:complexType name="PreChargeReqType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="DC_EVStatus" type="v2gci_t:DC_EVStatusType"/>
                <xs:element name="EVTargetVoltage" type="v2gci_t:PhysicalValueType"/>
                <xs:element name="EVTargetCurrent" type="v2gci_t:PhysicalValueType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="PreChargeRes" type="PreChargeResType" substitutionGroup="BodyElement"/>
<xs:complexType name="PreChargeResType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                <xs:element name="DC_EVSEStatus" type="v2gci_t:DC_EVSEStatusType"/>
                <xs:element name="EVSEPresentVoltage" type="v2gci_t:PhysicalValueType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!--                    -->
<!-- Current Demand -->
<!--                    -->
<xs:element name="CurrentDemandReq" type="CurrentDemandReqType" substitutionGroup="BodyElement"/>
<xs:complexType name="CurrentDemandReqType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="DC_EVStatus" type="v2gci_t:DC_EVStatusType"/>
                <xs:element name="EVTargetCurrent" type="v2gci_t:PhysicalValueType"/>
                <xs:element name="EVMaximumVoltageLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                <xs:element name="EVMaximumCurrentLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                <xs:element name="EVMaximumPowerLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                <xs:element name="BulkChargingComplete" type="xs:boolean" minOccurs="0"/>
                <xs:element name="ChargingComplete" type="xs:boolean"/>
                <xs:element name="RemainingTimeToFullSoC" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                <xs:element name="RemainingTimeToBulkSoC" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                <xs:element name="EVTargetVoltage" type="v2gci_t:PhysicalValueType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="CurrentDemandRes" type="CurrentDemandResType" substitutionGroup="BodyElement"/>
<xs:complexType name="CurrentDemandResType">
    <xs:complexContent>
        <xs:extension base="BodyBaseType">
            <xs:sequence>
                <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
```

```
                    <xs:element name="DC_EVSEStatus" type="v2gci_t:DC_EVSEStatusType"/>
                    <xs:element name="EVSEPresentVoltage" type="v2gci_t:PhysicalValueType"/>
                    <xs:element name="EVSEPresentCurrent" type="v2gci_t:PhysicalValueType"/>
                    <xs:element name="EVSECurrentLimitAchieved" type="xs:boolean"/>
                    <xs:element name="EVSEVoltageLimitAchieved" type="xs:boolean"/>
                    <xs:element name="EVSEPowerLimitAchieved" type="xs:boolean"/>
                    <xs:element name="EVSEMaximumVoltageLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                    <xs:element name="EVSEMaximumCurrentLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                    <xs:element name="EVSEMaximumPowerLimit" type="v2gci_t:PhysicalValueType"
minOccurs="0"/>
                    <xs:element name="EVSEID" type="v2gci_t:evseIDType"/>
                    <xs:element name="SASchedule TupleID" type="v2gci_t:SAIDType"/>
                    <xs:element name="MeterInfo" type="v2gci_t:MeterInfoType" minOccurs="0"/>
                    <xs:element name="ReceiptRequired" type="xs:boolean" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                -->
    <!-- Welding Detection -->
    <!--                -->
    <xs:element name="WeldingDetectionReq" type="WeldingDetectionReqType" substitutionGroup="BodyElement"/>
    <xs:complexType name="WeldingDetectionReqType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="DC_EVStatus" type="v2gci_t:DC_EVStatusType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="WeldingDetectionRes" type="WeldingDetectionResType" substitutionGroup="BodyElement"/>
    <xs:complexType name="WeldingDetectionResType">
        <xs:complexContent>
            <xs:extension base="BodyBaseType">
                <xs:sequence>
                    <xs:element name="ResponseCode" type="v2gci_t:responseCodeType"/>
                    <xs:element name="DC_EVSEStatus" type="v2gci_t:DC_EVSEStatusType"/>
                    <xs:element name="EVSEPresentVoltage" type="v2gci_t:PhysicalValueType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
</xs:schema>
```

F.6 V2G_CI_MsgDataTypes.xsd

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="urn:iec:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="urn:iec:15118:2:2013:MsgDataTypes" elementFormDefault="qualified
" attributeFormDefault="qualified">
    <xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="xmldsig-core-schema.xsd"/>
    <!-- ============================ -->
    <!-- Complex types                -->
    <!-- ============================ -->
    <!--                              -->
    <!-- service-related types        -->
    <!--                              -->
    <xs:complexType name="ServiceType">
```

```
        <xs:sequence>
            <xs:element name="ServiceID" type="serviceIDType"/>
            <xs:element name="ServiceName" type="serviceNameType" minOccurs="0"/>
            <xs:element name="ServiceCategory" type="serviceCategoryType"/>
            <xs:element name="ServiceScope" type="serviceScopeType" minOccurs="0"/>
            <xs:element name="FreeService" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ServiceListType">
        <xs:sequence>
            <xs:element name="Service" type="ServiceType" maxOccurs="8"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="SelectedServiceListType">
        <xs:sequence>
            <xs:element name="SelectedService" type="SelectedServiceType" maxOccurs="16"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="SelectedServiceType">
        <xs:sequence>
            <xs:element name="ServiceID" type="serviceIDType"/>
            <xs:element name="ParameterSetID" type="xs:short" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ServiceParameterListType">
        <xs:sequence>
            <xs:element name="ParameterSet" type="ParameterSetType" maxOccurs="255"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ParameterSetType">
        <xs:sequence>
            <xs:element name="ParameterSetID" type="xs:short"/>
            <xs:element name="Parameter" type="Parameter Type" maxOccurs="16"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ParameterType">
        <xs:choice>
            <xs:element name="boolValue" type="xs:boolean"/>
            <xs:element name="byteValue" type="xs:byte"/>
            <xs:element name="shortValue" type="xs:short"/>
            <xs:element name="intValue" type="xs:int"/>
            <xs:element name="physicalValue" type="PhysicalValueType"/>
            <xs:element name="stringValue" type="xs:string"/>
        </xs:choice>
        <xs:attribute name="Name" type="xs:string" use="required"/>
    </xs:complexType>
    <xs:complexType name="ChargeServiceType">
        <xs:complexContent>
            <xs:extension base="ServiceType">
                <xs:sequence>
                    <xs:element name="SupportedEnergyTransferMode" type="SupportedEnergyTransferModeType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="SupportedEnergyTransferModeType">
        <xs:sequence>
            <xs:element name="EnergyTransferMode" type="EnergyTransferModeType" maxOccurs="6"/>
        </xs:sequence>
    </xs:complexType>
    <!--                              -->
```

```
<!-- security related types -->
<!--                        -->
<xs:complexType name="ContractSignatureEncryptedPrivateKeyType">
    <xs:simpleContent>
        <xs:extension base="privateKeyType">
            <xs:attribute name="Id" type="xs:ID" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="DiffieHellmanPublickeyType">
    <xs:simpleContent>
        <xs:extension base="dhPublickeyType">
            <xs:attribute name="Id" type="xs:ID" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="EMAIDType">
    <xs:simpleContent>
        <xs:extension base="eMAIDType">
            <xs:attribute name="Id" type="xs:ID" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="CertificateChainType">
    <xs:sequence>
        <xs:element name="Certificate" type="certificateType"/>
        <xs:element name="SubCertificates" type="SubCertificatesType" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:ID"/>
</xs:complexType>
<xs:complexType name="SubCertificatesType">
    <xs:sequence>
        <xs:element name="Certificate" type="certificateType" maxOccurs="4"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ListOfRootCertificateIDsType">
    <xs:sequence>
        <xs:element name="RootCertificateID" type="xmldsig:X509IssuerSerialType" maxOccurs="20"/>
    </xs:sequence>
</xs:complexType>
<!--                                -->
<!-- metering related types -->
<!--                                -->
<xs:complexType name="MeterInfoType">
    <xs:sequence>
        <xs:element name="MeterID" type="meterIDType"/>
        <xs:element name="MeterReading" type="xs:unsignedLong" minOccurs="0"/>
        <xs:element name="SigMeterReading" type="sigMeterReadingType" minOccurs="0"/>
        <xs:element name="MeterStatus" type="meterStatusType" minOccurs="0"/>
        <xs:element name="TMeter" type="xs:long" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<!--                          -->
<!-- Physical value type -->
<!--                          -->
<xs:complexType name="PhysicalValueType">
    <xs:sequence>
        <xs:element name="Multiplier" type="unitMultiplierType"/>
        <xs:element name="Unit" type="unitSymbolType"/>
        <xs:element name="Value" type="xs:short"/>
    </xs:sequence>
```

```
</xs:complexType>
<!--                    -->
<!-- header related types -->
<!--                    -->
<xs:complexType name="NotificationType">
    <xs:sequence>
        <xs:element name="FaultCode" type="faultCodeType"/>
        <xs:element name="FaultMsg" type="faultMsgType" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<!--                    -->
<!-- Tariff related types -->
<!--                    -->
<xs:complexType name="SASchedulesType" abstract="true"/>
<xs:element name="SASchedules" type="SASchedulesType" abstract="true"/>
<xs:element name="SASchedulesList" type="SASchedulesListType" substitutionGroup="SASchedules"/>
<xs:complexType name="SASchedulesListType">
    <xs:complexContent>
        <xs:extension base="SASchedulesType">
            <xs:sequence>
                <xs:element name="SAScheduleTuple" type="SAScheduleTupleType" maxOccurs="3"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="SAScheduleTupleType">
    <xs:sequence>
        <xs:element name="SAScheduleTupleID" type="SAIDType"/>
        <xs:element name="PMaxSchedule" type="PMaxScheduleType"/>
        <xs:element name="SalesTariff" type="SalesTariffType" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="SalesTariffType">
    <xs:sequence>
        <xs:element name="SalesTariffID" type="SAIDType"/>
        <xs:element name="SalesTariffDescription" type="tariffDescriptionType" minOccurs="0"/>
        <xs:element name="NumEPriceLevels" type="xs:unsignedByte" minOccurs="0"/>
        <xs:element ref="SalesTariffEntry" maxOccurs="1024"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:ID"/>
</xs:complexType>
<xs:complexType name="PMaxScheduleType">
    <xs:sequence>
        <xs:element ref="PMaxScheduleEntry" maxOccurs="1024"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="Entry" type="EntryType" abstract="true"/>
<xs:complexType name="EntryType" abstract="true">
    <xs:sequence>
        <xs:element ref="TimeInterval"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="SalesTariffEntry" type="SalesTariffEntryType" substitutionGroup="Entry"/>
<xs:complexType name="SalesTariffEntryType">
    <xs:complexContent>
        <xs:extension base="EntryType">
            <xs:sequence>
                <xs:element name="EPriceLevel" type="xs:unsignedByte" minOccurs="0"/>
                <xs:element name="ConsumptionCost" type="ConsumptionCostType" minOccurs="0"
                maxOccurs="3"/>
            </xs:sequence>
```

```
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="PMaxScheduleEntry" type="PMaxScheduleEntryType" substitutionGroup="Entry"/>
    <xs:complexType name="PMaxScheduleEntryType">
        <xs:complexContent>
            <xs:extension base="EntryType">
                <xs:sequence>
                    <xs:element name="PMax" type="PhysicalValueType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="IntervalType" abstract="true"/>
    <xs:element name="TimeInterval" type="IntervalType" abstract="true"/>
    <xs:element name="RelativeTimeInterval" type="RelativeTimeIntervalType" substitutionGroup="TimeInterval"/>
    <xs:complexType name="RelativeTimeIntervalType">
        <xs:complexContent>
            <xs:extension base="IntervalType">
                <xs:sequence>
                    <xs:element name="start">
                        <xs:simpleType>
                            <xs:restriction base="xs:unsignedInt">
                                <xs:minInclusive value="0"/>
                                <xs:maxInclusive value="16777214"/>
                            </xs:restriction>
                        </xs:simpleType>
                    </xs:element>
                    <xs:element name="duration" minOccurs="0">
                        <xs:simpleType>
                            <xs:restriction base="xs:unsignedInt">
                                <xs:minInclusive value="0"/>
                                <xs:maxInclusive value="86400"/>
                            </xs:restriction>
                        </xs:simpleType>
                    </xs:element>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="ConsumptionCostType">
        <xs:sequence>
            <xs:element name="startValue" type="PhysicalValueType"/>
            <xs:element name="Cost" type="CostType" maxOccurs="3"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="CostType">
        <xs:sequence>
            <xs:element name="costKind" type="costKindType"/>
            <xs:element name="amount" type="xs:unsignedInt"/>
            <xs:element name="amountMultiplier" type="UnitMultiplierType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <!--                    -->
    <!-- EV/EVSE related types -->
    <!--                    -->
    <xs:complexType name="EVSEStatusType" abstract="true">
        <xs:sequence>
            <xs:element name="NotificationMaxDelay" type="xs:unsignedShort"/>
            <xs:element name="EVSENotification" type="EVSENotificationType"/>
        </xs:sequence>
```

```
</xs:complexType>
<xs:element name="EVSEStatus" type="EVSEStatusType" abstract="true"/>
<xs:element name="AC_EVSEStatus" type="AC_EVSEStatusType" substitutionGroup="EVSEStatus"/>
<xs:complexType name="AC_EVSEStatusType">
    <xs:complexContent>
        <xs:extension base="EVSEStatusType">
            <xs:sequence>
                <xs:element name="RCD" type="xs:boolean"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="EVStatusType" abstract="true"/>
<xs:element name="EVStatus" type="EVStatusType" abstract="true"/>
<xs:element name="DC_EVSEStatus" type="DC_EVSEStatusType" substitutionGroup="EVSEStatus"/>
<xs:complexType name="DC_EVSEStatusType">
    <xs:complexContent>
        <xs:extension base="EVSEStatusType">
            <xs:sequence>
                <xs:element name="EVSEIsolationStatus" type="IsolationLevelType" minOccurs="0"/>
                <xs:element name="EVSEStatusCode" type="DC_EVSEStatusCodeType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="DC_EVStatus" type="DC_EVStatusType" substitutionGroup="EVStatus"/>
<xs:complexType name="DC_EVStatusType">
    <xs:complexContent>
        <xs:extension base="EVStatusType">
            <xs:sequence>
                <xs:element name="EVReady" type="xs:boolean"/>
                <xs:element name="EVErrorCode" type="DC_EVErrorCodeType"/>
                <xs:element name="EVRESSSOC" type="percentValueType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!--                                    -->
<!-- EVSE/EV Charge Parameter related types -->
<!--                                    -->
<xs:complexType name="EVChargeParameterType" abstract="true">
    <xs:sequence>
        <xs:element name="DepartureTime" type="xs:unsignedInt" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="EVChargeParameter" type="EVChargeParameterType" abstract="true"/>
<xs:element name="AC_EVChargeParameter" type="AC_EVChargeParameterType"
substitutionGroup="EVChargeParameter"/>
<xs:complexType name="AC_EVChargeParameterType">
    <xs:complexContent>
        <xs:extension base="EVChargeParameterType">
            <xs:sequence>
                <xs:element name="EAmount" type="PhysicalValueType"/>
                <xs:element name="EVMaxVoltage" type="PhysicalValueType"/>
                <xs:element name="EVMaxCurrent" type="PhysicalValueType"/>
                <xs:element name="EVMinCurrent" type="PhysicalValueType"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="DC_EVChargeParameter" type="DC_EVChargeParameterType"
```

```
substitutionGroup="EVChargeParameter"/>
    <xs:complexType name="DC_EVChargeParameterType">
        <xs:complexContent>
            <xs:extension base="EVChargeParameterType">
                <xs:sequence>
                    <xs:element name="DC_EVStatus" type="DC_EVStatusType"/>
                    <xs:element name="EVMaximumCurrentLimit" type="PhysicalValueType"/>
                    <xs:element name="EVMaximumPowerLimit" type="PhysicalValueType" minOccurs="0"/>
                    <xs:element name="EVMaximumVoltageLimit" type="PhysicalValueType"/>
                    <xs:element name="EVEnergyCapacity" type="PhysicalValueType" minOccurs="0"/>
                    <xs:element name="EVEnergyRequest" type="PhysicalValueType" minOccurs="0"/>
                    <xs:element name="FullSOC" type="percentValueType" minOccurs="0"/>
                    <xs:element name="BulkSOC" type="percentValueType" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="EVSEChargeParameterType" abstract="true"/>
    <xs:element name="EVSEChargeParameter" type="EVSEChargeParameterType" abstract="true"/>
    <xs:element name="AC_EVSEChargeParameter" type="AC_EVSEChargeParameterType"
substitutionGroup="EVSEChargeParameter"/>
    <xs:complexType name="AC_EVSEChargeParameterType">
        <xs:complexContent>
            <xs:extension base="EVSEChargeParameterType">
                <xs:sequence>
                    <xs:element name="AC_EVSEStatus" type="AC_EVSEStatusType"/>
                    <!--<xs:element name="EVSEMaxVoltage" type="PhysicalValueType"/> -->
                    <xs:element name="EVSENominalVoltage" type="PhysicalValueType"/>
                    <xs:element name="EVSEMaxCurrent" type="PhysicalValueType"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:element name="DC_EVSEChargeParameter" type="DC_EVSEChargeParameterType"
substitutionGroup="EVSEChargeParameter"/>
    <xs:complexType name="DC_EVSEChargeParameterType">
        <xs:complexContent>
            <xs:extension base="EVSEChargeParameterType">
                <xs:sequence>
                    <xs:element name="DC_EVSEStatus" type="DC_EVSEStatusType"/>
                    <xs:element name="EVSEMaximumCurrentLimit" type="PhysicalValueType"/>
                    <xs:element name="EVSEMaximumPowerLimit" type="PhysicalValueType"/>
                    <xs:element name="EVSEMaximumVoltageLimit" type="PhysicalValueType"/>
                    <xs:element name="EVSEMinimumCurrentLimit" type="PhysicalValueType"/>
                    <xs:element name="EVSEMinimumVoltageLimit" type="PhysicalValueType"/>
                    <xs:element name="EVSECurrentRegulationTolerance" type="PhysicalValueType" minOccurs
                    ="0"/>
                    <xs:element name="EVSEPeakCurrentRipple" type="PhysicalValueType"/>
                    <xs:element name="EVSEEnergyToBeDelivered" type="PhysicalValueType" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <!--                                    -->
    <!-- EV Power Delivery related types   -->
    <!--                                    -->
    <xs:complexType name="EVPowerDeliveryParameterType" abstract="true"/>
    <xs:element name="EVPowerDeliveryParameter" type="EVPowerDeliveryParameterType" abstract="true"/>
    <xs:element name="DC_EVPowerDeliveryParameter" type="DC_EVPowerDeliveryParameterType"
substitutionGroup="EVPowerDeliveryParameter"/>
```

```
<xs:complexType name="DC_EVPowerDeliveryParameterType">
    <xs:complexContent>
        <xs:extension base="EVPowerDeliveryParameterType">
            <xs:sequence>
                <xs:element name="DC_EVStatus" type="DC_EVStatusType"/>
                <xs:element name="BulkChargingComplete" type="xs:boolean" minOccurs="0"/>
                <xs:element name="ChargingComplete" type="xs:boolean"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<!--                    -->
<!-- ChargingProfileType -->
<!--                    -->
<xs:complexType name="ChargingProfileType">
    <xs:sequence>
        <xs:element name="ProfileEntry" type="ProfileEntryType" maxOccurs="24"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ProfileEntryType">
    <xs:sequence>
        <xs:element name="ChargingProfileEntryStart" type="xs:unsignedInt"/>
        <!-- <xs:element name="ChargingProfileEntryMaxPower" type="PMaxType"/> -->
        <xs:element name="ChargingProfileEntryMaxPower" type="PhysicalValueType"/>
        <!-- 2013-06-20 VR: added per DE 101 and AT 5 -->
        <xs:element name="ChargingProfileEntryMaxNumberOfPhasesInUse" type="maxNumPhasesType"
        minOccurs="0"/>
        <!-- end add-->
    </xs:sequence>
</xs:complexType>
<!-- ========== ========== ========== -->
<!-- Simple types     -->
<!-- ========== ========== ========== -->
<!--            -->
<!-- General Types -->
<!--            -->
<xs:simpleType name="percentValueType">
    <xs:restriction base="xs:byte">
        <xs:minInclusive value="0"/>
        <xs:maxInclusive value="100"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="faultMsgType">
    <xs:restriction base="xs:string">
        <xs:maxLength value="64"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="EVSEProcessingType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Finished"/>
        <xs:enumeration value="Ongoing"/>
        <xs:enumeration value="Ongoing_WaitingForCustomerInteraction"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="EVSENotificationType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="StopCharging"/>
        <xs:enumeration value="ReNegotiation"/>
    </xs:restriction>
</xs:simpleType>
```

```
<xs:simpleType name="chargeProgressType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Start"/>
        <xs:enumeration value="Stop"/>
        <xs:enumeration value="Renegotiate"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="chargingSessionType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Terminate"/>
        <xs:enumeration value="Pause"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="serviceNameType">
    <xs:restriction base="xs:string">
        <xs:maxLength value="32"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="serviceCategoryType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="EVCharging"/>
        <xs:enumeration value="Internet"/>
        <xs:enumeration value="ContractCertificate"/>
        <xs:enumeration value="OtherCustom"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="serviceScopeType">
    <xs:restriction base="xs:string">
        <xs:maxLength value="64"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="maxNumPhasesType">
    <xs:restriction base="xs:byte">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="3"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="valueType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="bool"/>
        <xs:enumeration value="byte"/>
        <xs:enumeration value="short"/>
        <xs:enumeration value="int"/>
        <xs:enumeration value="physicalValue"/>
        <xs:enumeration value="string"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="meterStatusType">
    <xs:restriction base="xs:short"/>
</xs:simpleType>
<!--                    -->
<!-- EnergyTransferType -->
<!--                    -->
<xs:simpleType name="EnergyTransferModeType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="AC_single_phase_core"/>
        <xs:enumeration value="AC_three_phase_core"/>
        <xs:enumeration value="DC_core"/>
        <xs:enumeration value="DC_extended"/>
        <xs:enumeration value="DC_combo_core"/>
        <xs:enumeration value="DC_unique"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <!--                                    -->
    <!-- security types                     -->
    <!--                                    -->
    <xs:simpleType name="ganChallengeType">
        <xs:restriction base="xs:base64Binary">
            <xs:length value="16"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="certificateType">
        <xs:restriction base="xs:base64Binary">
            <xs:maxLength value="800"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="dHpublickeyType">
        <xs:restriction base="xs:base64Binary">
            <xs:maxLength value="65"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="privateKeyType">
        <xs:restriction base="xs:base64Binary">
            <xs:maxLength value="48"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="sigMeterReadingType">
        <xs:restriction base="xs:base64Binary">
            <xs:maxLength value="64"/>
        </xs:restriction>
    </xs:simpleType>
    <!--                                    -->
    <!-- Identification Numbers             -->
    <!--                                    -->
    <xs:simpleType name="sessionIDType">
        <xs:restriction base="xs:hexBinary">
            <xs:maxLength value="8"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="evccIDType">
        <xs:restriction base="xs:hexBinary">
            <xs:maxLength value="6"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="evseIDType">
        <xs:restriction base="xs:string">
            <xs:minLength value="7"/>
            <xs:maxLength value="37"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="serviceIDType">
        <xs:restriction base="xs:unsignedShort"/>
    </xs:simpleType>
    <xs:simpleType name="eMAIDType">
        <xs:restriction base="xs:string">
            <xs:minLength value="14"/>
            <xs:maxLength value="15"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="meterIDType">
        <xs:restriction base="xs:string">
            <xs:maxLength value="32"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <!--                    -->
    <!-- Tariffs and payment -->
    <!--                    -->
    <xs:simpleType name="SAIDType">
        <xs:restriction base="xs:unsignedByte">
            <xs:minInclusive value="1"/>
            <xs:maxInclusive value="255"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="tariffDescriptionType">
        <xs:restriction base="xs:string">
            <xs:maxLength value="32"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="costKindType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="relativePricePercentage"/>
            <xs:enumeration value="RenewableGenerationPercentage"/>
            <xs:enumeration value="CarbonDioxideEmission"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="PaymentOptionListType">
        <xs:sequence>
            <xs:element name="PaymentOption" type="paymentOptionType" minOccurs="1" maxOccurs="2"/>
        </xs:sequence>
    </xs:complexType>
    <xs:simpleType name="paymentOptionType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="Contract"/>
            <xs:enumeration value="ExternalPayment"/>
        </xs:restriction>
    </xs:simpleType>
    <!--                          -->
    <!-- Fault and Response Codes -->
    <!--                          -->
    <xs:simpleType name="faultCodeType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="ParsingError"/>
            <xs:enumeration value="NoTLSRootCertificatAvailable"/>
            <xs:enumeration value="UnknownError"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="responseCodeType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="OK"/>
            <xs:enumeration value="OK_NewSessionEstablished"/>
            <xs:enumeration value="OK_OldSessionJoined"/>
            <xs:enumeration value="OK_CertificatesExpiresSoon"/>
            <xs:enumeration value="FAILED"/>
            <xs:enumeration value="FAILED_SequenceError"/>
            <xs:enumeration value="FAILED_ServiceIDInvalid"/>
            <xs:enumeration value="FAILED_UnknownSession"/>
            <xs:enumeration value="FAILED_ServiceSelectionInvalid"/>
            <xs:enumeration value="FAILED_PaymentSelectionInvalid"/>
            <xs:enumeration value="FAILED_CertificateExpired"/>
            <xs:enumeration value="FAILED_SignatureError"/>
            <xs:enumeration value="FAILED_NoCertificatesAvailable"/>
            <xs:enumeration value="FAILED_CertChainError"/>
            <xs:enumeration value="FAILED_ChallengeInvalid"/>
```

```
            <xs:enumeration value="FAILED_ContractCanceled"/>
            <xs:enumeration value="FAILED_WrongChargeParameter"/>
            <xs:enumeration value="FAILED_PowerDeliveryNotApplied"/>
            <xs:enumeration value="FAILED_TariffSelectionInvalid"/>
            <xs:enumeration value="FAILED_ChargingProfileInvalid"/>
            <xs:enumeration value="FAILED_MeteringSignatureNotValid"/>
            <xs:enumeration value="FAILED_NoChargeServiceSelected"/>
            <xs:enumeration value="FAILED_WrongEnergyTransferMode"/>
            <xs:enumeration value="FAILED_ContactorError"/>
            <xs:enumeration value="FAILED_CertificateNotAllowedAtThisEVSE"/>
            <xs:enumeration value="FAILED_CertificateRevoked"/>
        </xs:restriction>
    </xs:simpleType>
    <!--                    -->
    <!-- Multiplier and Unit Types -->
    <!--                    -->
    <xs:simpleType name="unitMultiplierType">
        <xs:restriction base="xs:byte">
            <xs:minInclusive value="-3"/>
            <xs:maxInclusive value="3"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="unitSymbolType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="h">
                <xs:annotation>
                    <xs:documentation>Time in hours</xs:documentation>
                </xs:annotation>0
            </xs:enumeration>
            <xs:enumeration value="m">
                <xs:annotation>
                    <xs:documentation>Time in minutes</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="s">
                <xs:annotation>
                    <xs:documentation>Time in seconds</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="A">
                <xs:annotation>
                    <xs:documentation>Current in Ampere</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="V">
                <xs:annotation>
                    <xs:documentation>Voltage in Volt</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="W">
                <xs:annotation>
                    <xs:documentation>Active power in Watt</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="Wh">
                <xs:annotation>
                    <xs:documentation>Real energy in Watt hours</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
        </xs:restriction>
    </xs:simpleType>
```

```
        <!--                -->
        <!-- only DC related-->
        <!--                -->
        <xs:simpleType name="DC_EVSEStatusCodeType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="EVSE_NotReady"/>
                <xs:enumeration value="EVSE_Ready"/>
                <xs:enumeration value="EVSE_Shutdown"/>
                <xs:enumeration value="EVSE_UtilityInterruptEvent"/>
                <xs:enumeration value="EVSE_IsolationMonitoringActive"/>
                <xs:enumeration value="EVSE_EmergencyShutdown"/>
                <xs:enumeration value="EVSE_Malfunction"/>
                <xs:enumeration value="Reserved_8"/>
                <xs:enumeration value="Reserved_9"/>
                <xs:enumeration value="Reserved_A"/>
                <xs:enumeration value="Reserved_B"/>
                <xs:enumeration value="Reserved_C"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="IsolationLevelType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="Invalid"/>
                <xs:enumeration value="Valid"/>
                <xs:enumeration value="Warning"/>
                <xs:enumeration value="Fault"/>
                <xs:enumeration value="No_IMD"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="DC_EVErrorCodeType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="NO_ERROR"/>
                <xs:enumeration value="FAILED_RESSTemperatureInhibit"/>
                <xs:enumeration value="FAILED_EVShiftPosition"/>
                <xs:enumeration value="FAILED_ChargerConnectorLockFault"/>
                <xs:enumeration value="FAILED_EVRESSMalfunction"/>
                <xs:enumeration value="FAILED_ChargingCurrentDifferential"/>
                <xs:enumeration value="FAILED_ChargingVoltageOutOfRange"/>
                <xs:enumeration value="Reserved_A"/>
                <xs:enumeration value="Reserved_B"/>
                <xs:enumeration value="Reserved_C"/>
                <xs:enumeration value="FAILED_ChargingSystemIncompatibility"/>
                <xs:enumeration value="NoData"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:schema>
```

F.7 xmldsig-core-schema.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE schema PUBLIC "-//W3C//DTD XMLSchema 200102//EN" "http://www.w3.org/2001/XMLSchema.dtd" [
  <!ATTLIST schema
    xmlns:ds CDATA #FIXED "http://www.w3.org/2000/09/xmldsig#">
  <!ENTITY dsig "http://gost.e-ecolog.ru                    |dsig#">
  <!ENTITY % p ''>
  <!ENTITY % s ''>
 ]>
<!-- Schema for XML Signatures http://www.
    w3.org/2000/09/xmldsig#
    $Revision: 1.1 $ on $Date: 2002/02/08 20:32:26 $ by $Author: reagle $

    Copyright 2001 The Internet Society and W3C
    (Massachusetts Institute of Technology, Institut National
```

```
   de Recherche en Informatique et en Automatique, Keio
   University). All Rights Reserved. http://www.w3.org/
   Consortium/Legal/

   This document is governed by the W3C Software License [1]
   as described in the FAQ [2].

   [1] http://www.w3.org/Consortium/Legal/copyright-software-19980720
   [2] http://www.w3.org/Consortium/Legal/IPR-FAQ-20000620.html#DTD
-->
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="http://www.w3.org/2000/09/xmldsig#" version="0.1" elementFormDefault="qualified">

<!-- Basic Types Defined for Signatures -->

<simpleType name="CryptoBinary">
 <restriction base="base64Binary">
 </restriction>
</simpleType>

<!-- Start Signature -->

<element name="Signature" type=
"ds:SignatureType"/>
<complexType name="SignatureType">
 <sequence>
  <element ref="ds:SignedInfo"/>
  <element ref="ds:SignatureValue"/>
  <element ref="ds:KeyInfo" minOccurs="0"/>
  <element ref="ds:Object" minOccurs="0" maxOccurs="unbounded"/>
 </sequence>
 <attribute name="Id" type="ID" use="optional"/>
</complexType>

 <element name="SignatureValue" type="ds:SignatureValueType"/>
 <complexType name="SignatureValueType">
  <simpleContent>
   <extension base="base64Binary">
    <attribute name="Id" type="ID" use="optional"/>
   </extension>
  </simpleContent>
 </complexType>

<!-- Start SignedInfo -->

<element name="SignedInfo" type="ds:SignedInfoType"/>
<complexType name="SignedInfoType">
 <sequence>
  <element ref="ds:CanonicalizationMethod"/>
  <element ref="ds:SignatureMethod"/>
  <element ref="ds:Reference" maxOccurs="unbounded"/>
 </sequence>
 <attribute name="Id" type="ID" use="optional"/>
</complexType>

 <element name="CanonicalizationMethod" type="ds:CanonicalizationMethodType"/>
 <complexType name="CanonicalizationMethodType" mixed="true">
  <sequence>
   <any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
   <!-- (0,unbounded) elements from (1,1) namespace -->
  </sequence>
  <attribute name="Algorithm" type="anyURI" use="required"/>
 </complexType>

 <element name="SignatureMethod" type="ds:SignatureMethodType"/>
 <complexType name="SignatureMethodType" mixed="true">
```

```
    <sequence>
     <element name="HMACOutputLength" minOccurs="0" type="ds:HMACOutputLengthType"/>
     <any namespace="##other" minOccurs="0" maxOccurs="unbounded"/>
     <!-- (0,unbounded) elements from (1,1) external namespace -->
    </sequence>
    <attribute name="Algorithm" type="anyURI" use="required"/>
   </complexType>

   <!-- Start Reference -->
   <element name="Reference" type="ds:ReferenceType"/>
   <complexType name="ReferenceType">
    <sequence>
     <element ref="ds:Transforms" minOccurs="0"/>
     <element ref="ds:DigestMethod"/>
     <element ref="ds:DigestValue"/>
    </sequence>
    <attribute name="Id" type="ID" use="optional"/>
    <attribute name="URI" type="anyURI" use="optional"/>
    <attribute name="Type" type="anyURI" use="optional"/>
   </complexType>

    <element name="Transforms" type="ds:TransformsType"/>
    <complexType name="TransformsType">
     <sequence>
      <element ref="ds:Transform" maxOccurs="unbounded"/>
     </sequence>
    </complexType>

    <element name="Transform" type="ds:TransformType"/>
    <complexType name="TransformType" mixed="true">
     <choice minOccurs="0" maxOccurs="unbounded">
      <any namespace="##other" processContents="lax"/>
      <!-- (1,1) elements from (0,unbounded) namespace -->
      <element name="XPath" type="string"/>
     </choice>
     <attribute name="Algorithm" type="anyURI" use="required"/>
    </complexType>
   <!-- End Reference -->

   <element name="DigestMethod" type="ds:DigestMethodType"/>
   <complexType name="DigestMethodType" mixed="true">
    <sequence>
     <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="Algorithm" type="anyURI" use="required"/>
   </complexType>

   <element name="DigestValue" type="ds:DigestValueType"/>
   <simpleType name="DigestValueType">
    <restriction base="base64Binary"/>
   </simpleType>

   <!-- End SignedInfo -->

   <!-- Start KeyInfo -->
   <element name="KeyInfo" type="ds:KeyInfoType"/>
   <complexType name="KeyInfoType" mixed="true">
    <choice maxOccurs="unbounded">
     <element ref="ds:KeyName"/>
     <element ref="ds:KeyValue"/>
     <element ref="ds:RetrievalMethod"/>
     <element ref="ds:X509Data"/>
     <element ref="ds:PGPData"/>
     <element ref="ds:SPKIData"/>
```

```
    <element ref="ds:MgmtData"/>
    <any processContents="lax" namespace="##other"/>
    <!-- (1,1) elements from (0,unbounded) namespaces -->
   </choice>
   <attribute name="Id" type="ID" use="optional"/>
  </complexType>

  <element name="KeyName" type="string"/>
  <element name="MgmtData" type="string"/>

  <element name="KeyValue" type="ds:KeyValueType"/>
  <complexType name="KeyValueType" mixed="true">
   <choice>
    <element ref="ds:DSAKeyValue"/>
    <element ref="ds:RSAKeyValue"/>
    <any namespace="##other" processContents="lax"/>
   </choice>
  </complexType>

  <element name="RetrievalMethod" type="ds:RetrievalMethodType"/>
  <complexType name="RetrievalMethodType">
   <sequence>
    <element ref="ds:Transforms" minOccurs="0"/>
   </sequence>
   <attribute name="URI" type="anyURI"/>
   <attribute name="Type" type="anyURI" use="optional"/>
  </complexType>

<!-- Start X509Data -->

<element name="X509Data" type="ds:X509DataType"/>
<complexType name="X509DataType">
 <sequence maxOccurs="unbounded">
  <choice>
   <element name="X509IssuerSerial" type="ds:X509IssuerSerialType"/>
   <element name="X509SKI" type="base64Binary"/>
   <element name="X509SubjectName" type="string"/>
   <element name="X509Certificate" type="base64Binary"/>
   <element name="X509CRL" type="base64Binary"/>
   <any namespace="##other" processContents="lax"/>
  </choice>
 </sequence>
</complexType>

<complexType name="X509IssuerSerialType">
 <sequence>
  <element name="X509IssuerName" type="string"/>
  <element name="X509SerialNumber" type="integer"/>
 </sequence>
</complexType>

<!-- End X509Data -->

<!-- Begin PGPData -->

<element name="PGPData" type="ds:PGPDataType"/>
<complexType name="PGPDataType">
 <choice>
  <sequence>
   <element name="PGPKeyID" type="base64Binary"/>
   <element name="PGPKeyPacket" type="base64Binary" minOccurs="0"/>
   <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <sequence>
   <element name="PGPKeyPacket" type="base64Binary"/>
```

```
      <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </choice>
</complexType>

<!-- End PGPData -->

<!-- Begin SPKIData -->

<element name="SPKIData" type="ds:SPKIDataType"/>
<complexType name="SPKIDataType">
  <sequence maxOccurs="unbounded">
    <element name="SPKISexp" type="base64Binary"/>
    <any namespace="##other" processContents="lax" minOccurs="0"/>
  </sequence>
</complexType>

<!-- End SPKIData -->

<!-- End KeyInfo -->

<!-- Start Object (Manifest, SignatureProperty) -->

<element name="Object" type="ds:ObjectType"/>
<complexType name="ObjectType" mixed="true">
  <sequence minOccurs="0" maxOccurs="unbounded">
    <any namespace="##any" processContents="lax"/>
  </sequence>
  <attribute name="Id" type="ID" use="optional"/>
  <attribute name="MimeType" type="string" use="optional"/> <!-- add a grep facet -->
  <attribute name="Encoding" type="anyURI" use="optional"/>
</complexType>

<element name="Manifest" type="ds:ManifestType"/>
<complexType name="ManifestType">
  <sequence>
    <element ref="ds:Reference" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="Id" type="ID" use="optional"/>
</complexType>

<element name="SignatureProperties" type="ds:SignaturePropertiesType"/>
<complexType name="SignaturePropertiesType">
  <sequence>
    <element ref="ds:SignatureProperty" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="Id" type="ID" use="optional"/>
</complexType>

  <element name="SignatureProperty" type="ds:SignaturePropertyType"/>
  <complexType name="SignaturePropertyType" mixed="true">
    <choice maxOccurs="unbounded">
      <any namespace="##other" processContents="lax"/>
      <!-- (1,1) elements from (1,unbounded) namespaces -->
    </choice>
    <attribute name="Target" type="anyURI" use="required"/>
    <attribute name="Id" type="ID" use="optional"/>
  </complexType>

<!-- End Object (Manifest, SignatureProperty) -->

<!-- Start Algorithm Parameters -->

<simpleType name="HMACOutputLengthType">
  <restriction base="integer"/>
</simpleType>

<!-- Start KeyValue Element-types -->
```

```
    <element name="DSAKeyValue" type="ds:DSAKeyValueType"/>
    <complexType name="DSAKeyValueType">
     <sequence>
      <sequence minOccurs="0">
       <element name="P" type="ds:CryptoBinary"/>
       <element name="Q" type="ds:CryptoBinary"/>
      </sequence>
      <element name="G" type="ds:CryptoBinary" minOccurs="0"/>
      <element name="Y" type="ds:CryptoBinary"/>
      <element name="J" type="ds:CryptoBinary" minOccurs="0"/>
      <sequence minOccurs="0">
       <element name="Seed" type="ds:CryptoBinary"/>
       <element name="PgenCounter" type="ds:CryptoBinary"/>
      </sequence>
     </sequence>
    </complexType>

    <element name="RSAKeyValue" type="ds:RSAKeyValueType"/>
    <complexType name="RSAKeyValueType">
     <sequence>
      <element name="Modulus" type="ds:CryptoBinary"/>
      <element name="Exponent" type="ds:CryptoBinary"/>
     </sequence>
    </complexType>

    <!-- End KeyValue Element-types -->

    <!-- End Signature -->

</schema>
```

Приложение G
(обязательное)

Требования к идентификаторам

G.1 Идентификатор аккаунта EV (EMAID)

G.1.1 Синтаксис EVFID

Примечание — Идентификатор аккаунта EV (EMAID) представляет собой соответствующее поле данных в идентификаторе контракта (Contract ID), который описан в ISO 15118-1. В будущих редакциях настоящего стандарта он может быть переименован в идентификатор контракта, потому что описывает идентификатор контракта, а не идентификатор аккаунта. Часть данных «экземпляра аккаунта» (eMA Instance) может быть также переименована в «номер контракта».

EMAID должен соответствовать следующей структуре (обозначения соответствуют расширенной форме Бэкуса — Наура (ABNF), описанной в RFC 5234):

```
<ИАЭТС> = <код страны> <S> <идентификатор провайдера> <S> <экземпляр аккаунта> <S>
<контрольный символ>
<код страны> = 2 ALPHA
; двухзначный код страны в соответствии с ISO 3166-1 (Alpha-2-Code)
<идентификатор провайдера> = 3 (ALPHA / DIGIT)
; три буквенно-цифровых знака, которые определены и включены в список группой eMI3
<экземпляр аккаунта> = 9 (ALPHA / DIGIT)
; девять буквенно-цифровых знаков
<контрольный символ> = *1 (ALPHA / DIGIT)
; Дополнительно, но настоятельно рекомендуется, см. его вычисление в пункте H.1.3
ALPHA = %x41-5A / %x61-7A
; в соответствии с IETF RFC 5234(7-Bit ASCII)
DIGIT = %x30-39
; в соответствии с IETF RFC 5234 (7-Bit ASCII)
<S> = *1 ( "-" )
; произвольный разделительный знак
```

Примером действительного EMAID, таким образом, является «DE8AA1A2B3C4D59» или «DE-8AA-1A2B3C4D5-9» с тире.

G.1.2 Семантика EMAID

<EMAID> интерпретируется без учета регистра, т. е. «DE8AA1A2B3C4D59» является совершенно равнозначным идентификатору «de8aA1A2b3C4d59». Тире («-») может использоваться между элементами <код страны>, <идентификатор провайдера>, <экземпляр аккаунта> и <контрольный символ> при коммуникации с пользователями EV или EVSE для улучшения чтения, написания и печатания. Пример для иллюстрации: «DE-8AA-1A2B3C4D5-9». Если выбирается такая схема, разделительные знаки должны быть поставлены во всех трех местах. Каждый <EMAID> имеет длину не менее четырнадцати и не более пятнадцати знаков, исключая факультативные звездочки, или не менее семнадцати и не более восемнадцати знаков, включая факультативные разделительные знаки. В то время как <идентификатор провайдера> должен назначаться центральным выдающим органом (например, группа eMI3), каждый провайдер с присвоенным <идентификатором провайдера> может свободно выбирать <экземпляр аккаунта> в рамках вышеуказанных правил.

G.1.3 Расчет контрольного символа

Буквенно-цифровой символ рассчитывается с использованием первых 14 знаков <EMAID> и присоединяется на 15 позиции <EMAID>.

G.2 Идентификатор оборудования электропитания электрических транспортных средств (EVSEID)

G.2.1 Синтаксис EVSEID

<EVSEID> должен соответствовать следующей структуре (обозначения соответствуют расширенной форме Бэкуса — Наура (ABNF), описанной в IETF RFC 5234):

```
<EVSEID> = <код страны> <S> <идентификатор оператора EVSE> <S> <тип идентификатора>
<идентификатор точки отбора энергии>
<код страны> = 2 ALPHA
; двухзначный код страны, который соответствует следующим документам или их частям,
имеет нормативные ссылки в настоящем документе и необходим для его применения. Для дати-
рованных ссылок действует только редакция, на которую делается ссылка. Для недатированных
ссылок действует последняя редакция нормативного документа (с учетом всех изменений), на
который делается ссылка.
```

```
ISO 3166-1 (Alpha-2-Code)
<идентификатор оператора EVSE> = 3 (ALPHA / DIGIT)
; три буквенно-цифровых знака, которые определены и включены в список группой eMI3.
<тип идентификатора> = "E"
; один символ "E", обозначающий, что данный идентификатор представляет "EVSE".
<идентификатор точки отбора энергии> = (ALPHA / DIGIT) *30 (ALPHA / DIGIT / <S>)
; последовательность буквенно-цифровых знаков или разделительных знаков начинается
с буквенно-цифрового символа
ALPHA = %x41-5A / %x61-7A
; в соответствии с IETF RFC 5234 (7-Bit ASCII)
DIGIT = %x30-39
; в соответствии с IETF RFC 5234 (7-Bit ASCII)
<S> = *1 ( "*" )
; произвольный разделительный знак
```

Примером действительного EVSEID является «FR*A23*E45B*78C», где «FR» обозначает Францию, «A23» — конкретного оператора EVSE, «E» — тип «EVSE» и «45B*78C» — одну из точек отбора энергии.

Примечание — В отличие от EMAID, контрольный символ для EVSEID в настоящем стандарте не предусмотрен.

G.2.2 Семантика EVSEID

Каждый <EVSEID> имеет переменную длину и состоит не менее чем из семи знаков (двухзначный <код страны>, трехзначный <идентификатор оператора EVSE>, односимвольный <тип идентификатора>, односимвольный <идентификатор точки отбора энергии>) и не более чем из тридцати семи знаков (двухзначный <код страны>, трехзначный <идентификатор оператора EVSE>, односимвольный <тип идентификатора>, 31-символьный <идентификатор точки отбора энергии>). В то время как <идентификатор оператора EVSE> должен назначаться центральным выдающим органом, каждый оператор с присвоенным <идентификатором оператора EVSE> может свободно выбирать <идентификатор точки отбора энергии> в рамках вышеуказанных правил.

Примечание — В будущих редакциях настоящего стандарта будет возможным представление EVSEID без учета регистра во избежание неправильных написаний.

Приложение Н
(справочное)

Задание последовательности сообщений для пересогласования

Н.1 Обзор

В данном приложении даются примеры последовательностей сообщений, чтобы облегчить понимание механизма пересогласования.

В настоящем стандарте механизм пересогласования используется для изменения целевых настроек зарядки и графика во время цикла зарядки. Пересогласование может быть инициировано EVCC или SECC.

Для инициирования пересогласования SECC он должен установить EVSENotification в EVSEStatus в состояние «ReNegotiation». Если это выполнено, EVCC должен инициировать пересогласование в течение ограниченного времени (см. также **[V2G2-680]**).

Для инициирования пересогласования EVCC он может принять решение о выполнении пересогласования путем отправки PowerDeliveryReq с параметром ChargeProgress, установленным в состояние «Renegotiate» (см. также **[V2G2-522]**, **[V2G2-686]** и **[V2G2-683]**). EVCC может пересогласовать график зарядки следующим образом:

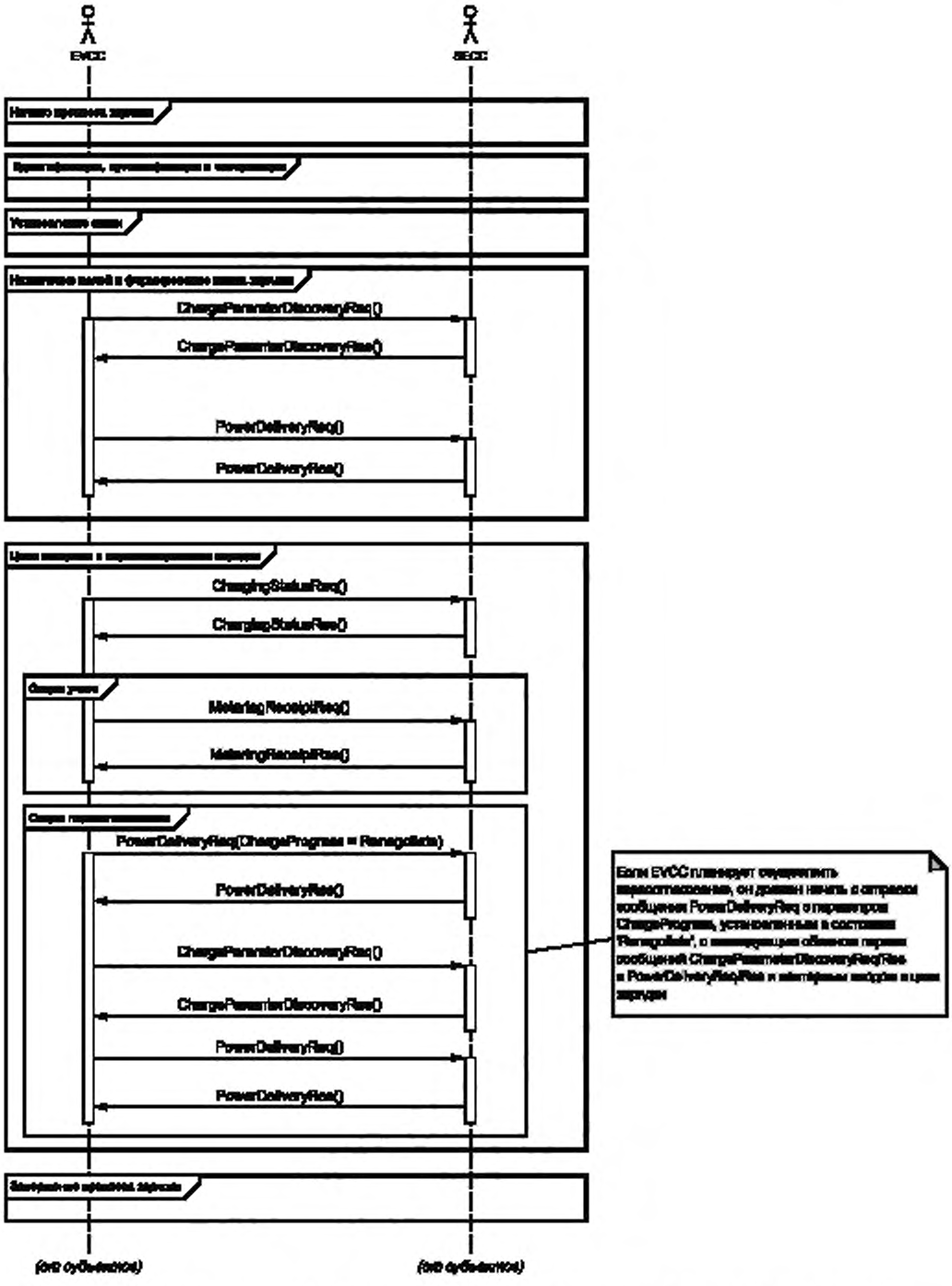


Рисунок Н.1 — Диаграмма последовательности для пересогласования, инициированного EVCC

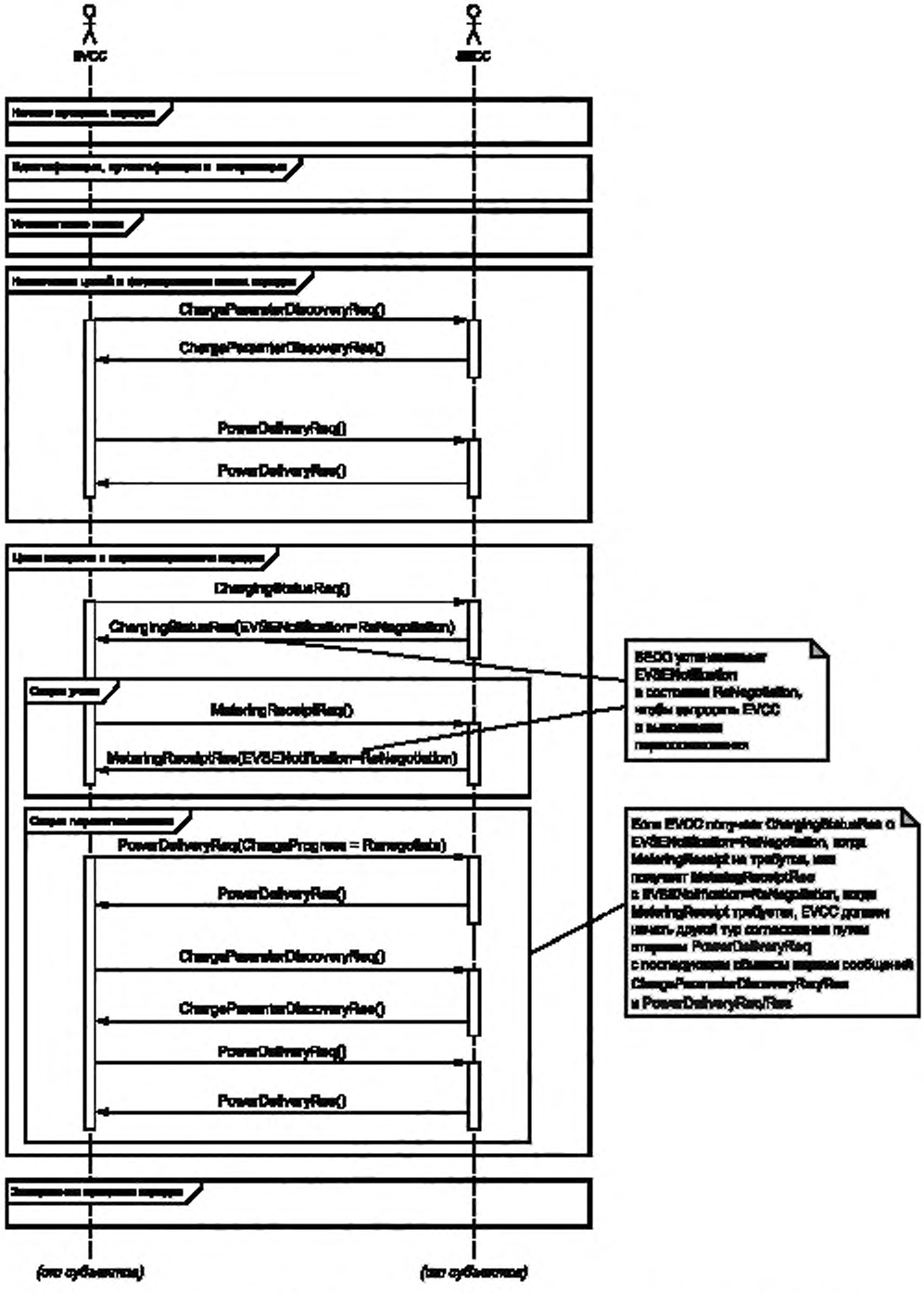


Рисунок Н.2 — Диаграмма последовательности для пересогласования, инициированного SECC

Н.2 Пересогласование после возобновления сеанса связи V2G

Процесс пересогласования разрешается только после предшествующего успешного согласования.

Если EVCC или SECC желает выполнить пересогласование после возобновления сеанса связи V2G, он должен сначала выполнить обмен сообщениями V2G до пары сообщений PowerDeliveryReq/Res и применить соответствующий механизм пересогласования, как описано в двух примерах, приведенных выше.

Приложение I
(справочное)

Привязка имен элементов сообщений ISO 15118 к терминам SAE J2847/2

I.1 Коды статуса SAE J2847/2

В таблицах I.1 и I.2 определены привязки имен перечисленных типов сигналов, используемых в контексте SAE J2847/2, и соответствующих имен элементов сообщений в ISO 15118. В общем случае термины «транспортное средство» и «зарядное оборудование» заменены сокращениями EV и EVSE соответственно.

Таблица I.1 — Привязка имен для кодов ошибок транспортного средства по SAE J2847/2

Обозначение	Определение SAE J2847/2 (код ошибки транспортного средства)	Определение ISO 15118
0x0	Транспортное средство не готово	DC_EVStatus: EVReady (Value = false)
0x1	Зарядка транспортного средства или передача энергии	DC_EVStatus: EVReady (Value = true)
0x2	Защита от перегрева перезаряжаемой энергоаккумулирующей системы	DC_EVStatus: DC_EVErrorCode (Value = FAILED_RESSTemperatureInhibit)
0x3	Положение контроллера движения транспортного средства	DC_EVStatus: DC_EVErrorCode (Value = FAILED_EVShiftPosition)
0x4	Неисправность замка соединителя зарядной станции	DC_EVStatus: DC_EVErrorCode (Value = FAILED_ChargerConnectorLockFault)
0x7	Неисправность перезаряжаемой энергоаккумулирующей системы транспортного средства	DC_EVStatus: DC_EVErrorCode (Value = FAILED_EVRESSMalfunction)
0x8	Перепад зарядного тока	DC_EVStatus: DC_EVErrorCode (Value = FAILED_ChargingCurrentdifferential)
0x9	Зарядное напряжение за пределами диапазона	DC_EVStatus: DC_EVErrorCode (Value = FAILED_ChargingVoltageOutOfRange)
0xA—0xC	Зарезервировано	DC_EVStatus: DC_EVErrorCode (Value = Reserved_A .. Reserved_C)
0xD	Несовместимость зарядной системы	DC_EVStatus: DC_EVErrorCode (Value = FAILED_ChargingSystemIncompatibility)
0xE	Другие несоответствия транспортного средства	См. значения ResponseCode
0xF	Данные отсутствуют	DC_EVStatus: DC_EVErrorCode (Value = NoData)

Примечание — Передаваемые данные соответствуют кодам статуса транспортного средства, описанным в SAE J2847/2.

Таблица I.2 — Привязка имен для кодов статуса зарядной станции по SAE J2847/2

Обозначение	Определение SAE J2847/2 (код статуса зарядной станции)	Определение ISO 15118
0x0	Зарядная станция в режиме ожидания/не готова	DC_EVSEStatusCode (value = FAILED_NotReady)
0x1	Зарядная станция готова/заряжает	DC_EVSEStatusCode (value = OK_Charging)
0x2	Превышены предоплаченные пределы зарядной станции	responseCode (value = FAILED_ContractCanceled)
0x3	Выключение зарядной станции	DC_EVSEStatusCode (value = EVSE_Shutdown)
0x4	Перебой энергоснабжения	DC_EVSEStatusCode (value = EVSE_UtilityInterruptEvent)
0x5	Внутренний мониторинг изоляции	Не применяется
0x6	Мониторинг изоляции активен	DC_EVSEStatusCode (value = EVSE_IsolationMonitoringActive)
0x7	Аварийное выключение зарядной станции	DC_EVSEStatusCode (value = EVSE_EVSE_EmergencyShutdown)
0x8 — 0xC	Зарезервировано	DC_EVSEStatusCode (value = Reserved_8 .. Reserved_C)
0xD	Несовместимость зарядной системы	responseCode (value = FAILED_WrongChargeParameter)
0xE	Неисправность зарядной станции	responseCode (value = EVSE_Malfunction)
0xF	Данные отсутствуют	Не применяется

Примечание 2 — Передаваемые данные соответствуют кодам статуса зарядной станции, описанным в SAE J2847/2.

I.2 Типы передачи энергии по SAE J2847/2

В таблице I.3 определена привязка имен для запрошенного транспортным средством типа передачи энергии по SAE J2847/2 и поддерживаемого зарядной станцией типа передачи энергии.

Таблица I.3 — Привязка имен для запрошенного транспортным средством типа передачи энергии по SAE J2847/2

Обозначение	Определение SAE J2847/2 (запрошенный транспортным средством тип передачи энергии)	Определение ISO 15118 (RequestedEnergyTransferMode)
0x00	(зарезервировано для) Зарядка переменным однофазным током типа 1/типа 2 через основные контакты	AC_single_phase_core
0x01	(зарезервировано для) Зарядка трехфазным переменным током типа 2 через основные контакты типа 2	AC_three_phase_core
0x02	Зарядка постоянным током типа 1/типа 2 через основные контакты	DC_core
0x03	Зарядка постоянным током комбо 1/комбо 2 с использованием дополнительных контактов	DC_extended

Окончание таблицы I.3

Обозначение	Определение SAE J2847/2 (запрошенный транспортным средством тип передачи энергии)	Определение ISO 15118 (RequestedEnergyTransferMode)
0x04	Зарядка постоянным током комбо 1/комбо 2 через основные контакты	DC_combo_core
0x05	(зарезервировано для) Специализированная зарядка постоянным током через специальный соединитель	DC_unique
0x08—0x0E	Зарезервировано для будущего использования	Reserved_8.. Reserved_E
0x0F	Не определено	Не определено

I.3 Сигналы SAE J2847/2

В таблице I.4 определена привязка имен для сигналов SAE J2847/2 к элементам сообщений ISO 15118.

Таблица I.4 — Привязка имен для сигналов SAE J2847/2 к элементам сообщений ISO 15118

Определение сигнала SAE J2847/2	Определение элемента сообщений ISO 15118
Основная зарядка завершена	DC_EVPowerDeliveryParameter: BulkChargingComplete CurrentDemandRequest: BulkChargingComplete
Состояние основной зарядки	DC_EVChargeParameter: BulkSOC
Запрос зарядного тока	CurrentDemandRequest: EVTargetCurrent
Достигнут предельный ток зарядной станции	CurrentDemandRes: EVSECurrentLimitAchieved
Допуск регулирования тока зарядной станции	DC_EVSEChargeParameter: EVSECurrentRegulationTolerance
Энергия зарядной станции, подлежащая передаче	DC_EVSEChargeParameter: EVSEEnergyToBeDelivered
Предел максимального тока зарядной станции	DC_EVSEChargeParameter: EVSEMaximumCurrentLimit CurrentDemandRes: EVSEMaximumCurrentLimit
Предел максимальной мощности зарядной станции	DC_EVSEChargeParameter: EVSEMaximumPowerLimit CurrentDemandRes: EVSEMaximumCurrentLimit
Предел максимального напряжения зарядной станции	DC_EVSEChargeParameter: EVSEMaximumVoltageLimit CurrentDemandRes: EVSEMaximumCurrentLimit
Предел минимального тока зарядной станции	DC_EVSEChargeParameter: EVSEMinimumCurrentLimit
Предел минимального напряжения зарядной станции	DC_EVSEChargeParameter: EVSEMinimumVoltageLimit
Пиковая пульсация тока зарядной станции	DC_EVSEChargeParameter: EVSEPeakCurrentRipple

Окончание таблицы I.4

Определение сигнала SAE J2847/2	Определение элемента сообщения ISO 15118
Достигнут предел мощности зарядной станции	CurrentDemandRes: EVSEPowerLimitAchieved
Статус зарядной станции	DC_EVSEStatus
Достигнут предел напряжения зарядной станции	CurrentDemandRes: EVSEVoltageLimitAchieved
Выходной ток	CurrentDemandRes: EVSEPresentCurrent
Соединитель заблокирован	Не применяется
Состояние полной зарядки	DC_EVChargeParameter: FullSOC
Энергетическая емкость EV	DC_EVChargeParameter: EVEnergyCapacity
Запрос энергии транспортным средством	DC_EVChargeParameter: EVEnergyRequest
Предел максимального тока транспортного средства	DC_EVChargeParameter: EVMaximumCurrentLimit
Предел максимальной мощности транспортного средства	DC_EVChargeParameter: EVMaximumPowerLimit
Предел максимального напряжения транспортного средства	DC_EVChargeParameter: EVMaximumVoltageLimit
Состояние зарядки перезаряжаемой энергоаккумулирующей системы транспортного средства	DC_EVStatusType: EVRESSSOC
Код ошибки транспортного средства	DC_EVErrorCode
Выходное напряжение	CurrentDemandRes: EVSEPresentVoltage

Приложение J
(справочное)

Примеры сообщений

J.1 Выбор дополнительной услуги

Экземпляры XML, приведенные в данном приложении, показывают примеры, как может быть реализован выбор дополнительных услуг. Предполагается следующее исходное условие: EVCC хочет использовать доступ к Интернету с протоколами HTTP и FTP, если это возможно.

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>3031323334353637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ServiceDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:PaymentOptionList>
                <v2gci_t:PaymentOption>Contract</v2gci_t:PaymentOption>
                <v2gci_t:PaymentOption>ExternalPayment</v2gci_t:PaymentOption>
            </v2gci_b:PaymentOptionList>
            <v2gci_b:ChargeService>
                <v2gci_t:ServiceID>1</v2gci_t:ServiceID>
                <v2gci_t:ServiceName>AC_DC_Charging</v2gci_t:ServiceName>
                <v2gci_t:ServiceCategory>EVCharging</v2gci_t:ServiceCategory>
                <v2gci_t:FreeService>true</v2gci_t:FreeService>
                <v2gci_t:SupportedEnergyTransferMode>
                    <v2gci_t:EnergyTransferMode>AC_three_phase_core</v2gci_t:EnergyTransferMode>
                    <v2gci_t:EnergyTransferMode>DC_extended</v2gci_t:EnergyTransferMode>
                </v2gci_t:SupportedEnergyTransferMode>
            </v2gci_b:ChargeService>
            <v2gci_b:ServiceList>
                <v2gci_t:Service>
                    <v2gci_t:ServiceID>5</v2gci_t:ServiceID>
                    <v2gci_t:ServiceName>Fast Internet</v2gci_t:ServiceName>
                    <v2gci_t:ServiceCategory>Internet</v2gci_t:ServiceCategory>
                    <v2gci_t:FreeService>false</v2gci_t:FreeService>
                </v2gci_t:Service>
                <v2gci_t:Service>
                    <v2gci_t:ServiceID>2</v2gci_t:ServiceID>
                    <v2gci_t:ServiceName>Certificate</v2gci_t:ServiceName>
                    <v2gci_t:ServiceCategory>ContractCertificate</v2gci_t:ServiceCategory>
                    <v2gci_t:FreeService>false</v2gci_t:FreeService>
                </v2gci_t:Service>
            </v2gci_b:ServiceList>
        </v2gci_b:ServiceDiscoveryRes>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 6 сообщения V2G — Сообщение ServiceDiscoveryRes

EVCC запрашивает подробности, касающиеся доступа к Интернету и обработки сертификата, сообщением ServiceDetailsReq. Пара сообщений ServiceDetailReq/Res для услуги доступа к Интернету выглядит следующим образом:

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>90813223934353837</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ServiceDetailReq>
            <v2gci_b:ServiceID>2</v2gci_b:ServiceID>
        </v2gci_b:ServiceDetailReq>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 7 сообщения V2G — Сообщение ServiceDiscoveryReq

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>90813223934353837</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ServiceDetailRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:ServiceID>2</v2gci_b:ServiceID>
            <v2gci_b:ServiceParameterList>
                <v2gci_t:ParameterSet>
                    <v2gci_t:ParameterSetID>3</v2gci_t:ParameterSetID>
                    <v2gci_t:Parameter v2gci_t:Name="Protocol">
                        <v2gci_t:stringValue>HTTP</v2gci_t:stringValue>
                    </v2gci_t:Parameter>
                    <v2gci_t:Parameter v2gci_t:Name="Port">
                        <v2gci_t:intValue>80</v2gci_t:intValue>
                    </v2gci_t:Parameter>
                </v2gci_t:ParameterSet>
                <v2gci_t:ParameterSet>
                    <v2gci_t:ParameterSetID>4</v2gci_t:ParameterSetID>
                    <v2gci_t:Parameter v2gci_t:Name="Protocol">
                        <v2gci_t:stringValue>HTTPS</v2gci_t:stringValue>
                    </v2gci_t:Parameter>
                    <v2gci_t:Parameter v2gci_t:Name="Port">
                        <v2gci_t:intValue>443</v2gci_t:intValue>
                    </v2gci_t:Parameter>
                </v2gci_t:ParameterSet>
            </v2gci_b:ServiceParameterList>
        </v2gci_b:ServiceDetailRes>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 8 сообщения V2G — Сообщение ServiceDetailsRes

Сообщение-ответ показывает, что SECC предлагает доступ к Интернету через HTTP с использованием порта 80 и через HTTPS с использованием порта 443, и поэтому EVCC не разрешается использовать FTP (идентификатор набора параметров = 1 или 2), а требуется использовать HTTP или HTTPS.

Далее EVCC запрашивает желаемые услуги с использованием PaymentServiceSelectionReq

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9031823334353837</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:PaymentServiceSelectionReq>
            <v2gci_b:SelectedPaymentOption>Contract</v2gci_b:SelectedPaymentOption>
            <v2gci_b:SelectedServiceList>
                <v2gci_t:SelectedService>
                    <v2gci_t:ServiceID>1</v2gci_t:ServiceID>
                    <!-- charge service -->
                </v2gci_t:SelectedService>
                <v2gci_t:SelectedService>
                    <v2gci_t:ServiceID>3</v2gci_t:ServiceID>
                    <!-- Internet service -->
                    <v2gci_t:ParameterSetID>3</v2gci_t:ParameterSetID>
                </v2gci_t:SelectedService>
            </v2gci_b:SelectedServiceList>
        </v2gci_b:PaymentServiceSelectionReq>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 9 сообщения V2G — Сообщение PaymentServiceSelectionReq

J.2 Примеры EXI-кодированных сообщений

Ниже приведены три примера (SessionSetupRes, ChargeParameterDiscoveryReq и CurrentDemandReq) сообщений V2G в виде простого XML-текста и эквивалентное представление EXI-формата.

J.2.1 Сообщение SessionSetupRes

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://gost.e-ecolog.ru                    ichema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9031823334353837</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:SessionSetupRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:EVSEID>FRA23E45878C</v2gci_b:EVSEID>
        </v2gci_b:SessionSetupRes>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 10 сообщения V2G — Представление сообщения SessionSetupRes в виде простого XML-текста

```
80 98 02 0C 0C 4C 8C CD 0D 4D 8D D1 E0 00 39 19 49 04 C8 CD 14 D0 D5 08 DC E1 0C 80
```

Пример 11 сообщения V2G — Представление сообщения SessionSetupRes в виде потока EXI-данных

J.2.2 Сообщение ChargeParameterDiscoveryReq (на основе зарядки переменным током)

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>3031323334353637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryReq>
            <v2gci_b:RequestedEnergyTransferMode>AC_single_phase_core</v2gci_b:RequestedEnergyTransferMode>
            <v2gci_t:AC_EVChargeParameter>
                <v2gci_t:DepartureTime>100</v2gci_t:DepartureTime>
                <v2gci_t:EAmount>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>Wh</v2gci_t:Unit>
                    <v2gci_t:Value>15</v2gci_t:Value>
                </v2gci_t:EAmount>
                <v2gci_t:EVMaxVoltage>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>V</v2gci_t:Unit>
                    <v2gci_t:Value>230</v2gci_t:Value>
                </v2gci_t:EVMaxVoltage>
                <v2gci_t:EVMaxCurrent>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>A</v2gci_t:Unit>
                    <v2gci_t:Value>32</v2gci_t:Value>
                </v2gci_t:EVMaxCurrent>
                <v2gci_t:EVMinCurrent>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>A</v2gci_t:Unit>
                    <v2gci_t:Value>0</v2gci_t:Value>
                </v2gci_t:EVMinCurrent>
            </v2gci_t:AC_EVChargeParameter>
        </v2gci_b:ChargeParameterDiscoveryReq>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 12 сообщения V2G — Представление в виде прозрачного XML-текста сообщения ChargeParameterDiscoveryReq (на основе зарядки переменным током)

```
80 98 02 0C 0C 4C 8C CD 0D 4D 8D D0 94 00 64 0C 30 09 01 88 1C C0 20 61 81 00 18 60 00 00
```

Пример 13 сообщения V2G — Представление сообщения ChargeParameterDiscoveryReq в виде потока EXI-данных

J.2.3 Сообщение CurrentDemandReq

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>3031323334353637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
```

```
            <v2gci_b:CurrentDemandReq>
                <v2gci_b:DC_EVStatus>
                    <v2gci_t:EVReady>true</v2gci_t:EVReady>
                    <v2gci_t:EVErrorCode>NO_ERROR</v2gci_t:EVErrorCode>
                    <v2gci_t:EVRESSSOC>55</v2gci_t:EVRESSSOC>
                </v2gci_b:DC_EVStatus>
                <v2gci_b:EVTargetCurrent>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>A</v2gci_t:Unit>
                    <v2gci_t:Value>60</v2gci_t:Value>
                </v2gci_b:EVTargetCurrent>
                <v2gci_b:ChargingComplete>false</v2gci_b:ChargingComplete>
                <v2gci_b:EVTargetVoltage>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>V</v2gci_t:Unit>
                    <v2gci_t:Value>450</v2gci_t:Value>
                </v2gci_b:EVTargetVoltage>
            </v2gci_b:CurrentDemandReq>
        </v2gci_d:Body>
    </v2gci_d:V2G_Message>
```

Пример 14 сообщения V2G — Представление сообщения CurrentDemandReq в виде простого XML-текста

80 98 02 0C 0C 4C 8C CD 0D 4D 8D D0 D1 00 1B 81 86 07 84 10 C4 0C 20 30 00

Пример 15 сообщения V2G — Представление сообщения CurrentDemandReq в виде потока EXI-данных

J.3 Информация о графиках и тарифах

J.3.1 Обзор

В настоящем пункте дается обзор лучшей практики реализации разных тарифных моделей в рамках схемы ChargeParameterDiscovery Req/Res. Даны следующие примеры:

- динамический график электросети GridSchedule без тарифа SalesTariff интерфейса связи V2G ISO 15118 (см. J.3.2);
- тариф на основе «времени использования» с постоянным значением для GridSchedule (см. J.3.3);
- тариф на основе «времени использования» с динамическим GridSchedule (см. J.3.4);
- тариф на основе «потребления» с постоянным значением для GridSchedule (см. J.3.5);
- несколько тарифов с разными пределами запроса в GridSchedule (см. J.3.6);
- тарифы на основе «времени использования» с relativePricePercentage (см. J.3.7).

Примечание 1 — Во всех следующих примерах показан элемент *тела* сообщения-ответа *ChargeParameterDiscovery* для зарядки переменным током.

Примечание 2 — Список примеров, который дан в настоящем приложении, не является исчерпывающим. Возможны другие варианты и комбинации информации о графике и тарифе на основе соответствующей модели бизнеса.

J.3.2 Динамический график электросети GridSchedule без тарифа SalesTariff интерфейса связи V2G ISO 15118

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<v2gci_d:Header>
    <v2gci_h:SessionID>303132333435363738</v2gci_h:SessionID>
</v2gci_d:Header>
<v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
```

```
<v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
<v2gci_t:SAScheduleList>
    <v2gci_t:SAScheduleTuple>
        <v2gci_t:SAScheduleTupleID>65</v2gci_t:SAScheduleTupleID>
        <v2gci_t:PMaxSchedule>
            <v2gci_t:PMaxScheduleEntry>
                <v2gci_t:RelativeTimeInterval>
                    <v2gci_t:start>0</v2gci_t:start>
                </v2gci_t:RelativeTimeInterval>
                <v2gci_t:PMax>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>W</v2gci_t:Unit>
                    <v2gci_t:Value>9600</v2gci_t:Value>
                </v2gci_t:PMax>
            </v2gci_t:PMaxScheduleEntry>
            <v2gci_t:PMaxScheduleEntry>
                <v2gci_t:RelativeTimeInterval>
                    <v2gci_t:start>2341</v2gci_t:start>
                </v2gci_t:RelativeTimeInterval>
                <v2gci_t:PMax>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>W</v2gci_t:Unit>
                    <v2gci_t:Value>22000</v2gci_t:Value>
                </v2gci_t:PMax>
            </v2gci_t:PMaxScheduleEntry>
            <v2gci_t:PMaxScheduleEntry>
                <v2gci_t:RelativeTimeInterval>
                    <v2gci_t:start>9541</v2gci_t:start>
                </v2gci_t:RelativeTimeInterval>
                <v2gci_t:PMax>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>W</v2gci_t:Unit>
                    <v2gci_t:Value>9600</v2gci_t:Value>
                </v2gci_t:PMax>
            </v2gci_t:PMaxScheduleEntry>
            <v2gci_t:PMaxScheduleEntry>
                <v2gci_t:RelativeTimeInterval>
                    <v2gci_t:start>23941</v2gci_t:start>
                </v2gci_t:RelativeTimeInterval>
                <v2gci_t:PMax>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>W</v2gci_t:Unit>
                    <v2gci_t:Value>22000</v2gci_t:Value>
                </v2gci_t:PMax>
            </v2gci_t:PMaxScheduleEntry>
            <v2gci_t:PMaxScheduleEntry>
                <v2gci_t:RelativeTimeInterval>
                    <v2gci_t:start>31141</v2gci_t:start>
                    <v2gci_t:duration>7200</v2gci_t:duration>
                </v2gci_t:RelativeTimeInterval>
                <v2gci_t:PMax>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>W</v2gci_t:Unit>
                    <v2gci_t:Value>12000</v2gci_t:Value>
                </v2gci_t:PMax>
            </v2gci_t:PMaxScheduleEntry>
        </v2gci_t:PMaxSchedule>
    </v2gci_t:SAScheduleTuple>
</v2gci_t:SAScheduleList>
<v2gci_t:AC_EVSEChargeParameter>
    <v2gci_t:AC_EVSEStatus>
```

```
                    <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
                    <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
                    <v2gci_t:RCD>true</v2gci_t:RCD>
                </v2gci_t:AC_EVSEStatus>
                <v2gci_t:EVSENominalVoltage>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>V</v2gci_t:Unit>
                    <v2gci_t:Value>230</v2gci_t:Value>
                </v2gci_t:EVSENominalVoltage>
                <v2gci_t:EVSEMaxCurrent>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>A</v2gci_t:Unit>
                    <v2gci_t:Value>50</v2gci_t:Value>
                </v2gci_t:EVSEMaxCurrent>
            </v2gci_t:AC_EVSEChargeParameter>
        </v2gci_b:ChargeParameterDiscoveryRes>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 18 сообщения V2G — Пример сообщения-ответа ChargeParameterDiscoveryRes для GridSchedule без SalesTariff

J.3.3 Тариф на основе «времени использования» с постоянным значением для GridSchedule

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iec:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iec:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iec:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iec:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<v2gci_d:Header>
    <v2gci_h:SessionID>3031323334353637</v2gci_h:SessionID>
</v2gci_d:Header>
<v2gci_d:Body>
    <v2gci_b:ChargeParameterDiscoveryRes>
        <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
        <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
        <v2gci_t:SAScheduleList>
            <v2gci_t:SAScheduleTuple>
                <v2gci_t:SAScheduleTupleID>55</v2gci_t:SAScheduleTupleID>
                <v2gci_t:PMaxSchedule>
                    <v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                            <v2gci_t:duration>30000</v2gci_t:duration>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:PMax>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>6600</v2gci_t:Value>
                        </v2gci_t:PMax>
                    </v2gci_t:PMaxScheduleEntry>
                </v2gci_t:PMaxSchedule>
                <v2gci_t:SalesTariff v2gci_t:id="ID001">
                    <v2gci_t:SalesTariffID>67</v2gci_t:SalesTariffID>
                    <v2gci_t:SalesTariffDescription>SalesTariffDescription1</v2gci_t:SalesTariffDescription>
                    <v2gci_t:NumEPriceLevels>3</v2gci_t:NumEPriceLevels>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
```

```
                    <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>2147</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>9674</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>3</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>14837</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>24431</v2gci_t:start>
                            <v2gci_t:duration>90000</v2gci_t:duration>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
            </v2gci_t:SalesTariff>
        </v2gci_t:SAScheduleTuple>
    </v2gci_t:SAScheduleList>
    <v2gci_t:AC_EVSEChargeParameter>
        <v2gci_t:AC_EVSEStatus>
            <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
            <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
            <v2gci_t:RCD>false</v2gci_t:RCD>
        </v2gci_t:AC_EVSEStatus>
        <v2gci_t:EVSENominalVoltage>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>

            <v2gci_t:Unit>V</v2gci_t:Unit>
            <v2gci_t:Value>230</v2gci_t:Value>
        </v2gci_t:EVSENominalVoltage>
        <v2gci_t:EVSEMaxCurrent>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
            <v2gci_t:Unit>A</v2gci_t:Unit>
            <v2gci_t:Value>60</v2gci_t:Value>
        </v2gci_t:EVSEMaxCurrent>
    </v2gci_t:AC_EVSEChargeParameter>
</v2gci_b:ChargeParameterDiscoveryRes>
</v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 17 сообщения V2G — Пример сообщения-ответа ChargeParameterDiscoveryRes для тарифа на основе «времени использования» с постоянным значением для GridSchedule

J.3.4 Тариф на основе «времени использования» с динамическим GridSchedule

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
```

```
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>8081323834353637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
            <v2gci_t:SAScheduleList>
                <v2gci_t:SAScheduleTuple>
                    <v2gci_t:SAScheduleTupleID>55</v2gci_t:SAScheduleTupleID>
                    <v2gci_t:PMaxSchedule>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>6600</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>2341</v2gci_t:start>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>22000</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>9541</v2gci_t:start>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>6600</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>2394</v2gci_t:start>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>22000</v2gci_t:Value>

                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>31141</v2gci_t:start>
                                <v2gci_t:duration>36000</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
```

```
                        <v2gci_t:Unit>W</v2gci_t:Unit>
                        <v2gci_t:Value>12000</v2gci_t:Value>
                    </v2gci_t:PMax>
                </v2gci_t:PMaxScheduleEntry>
            </v2gci_t:PMaxSchedule>
            <v2gci_t:SalesTariff v2gci_t:id="ID001">
                <!-- used by XML Signature Framework for signature assignment in msg header -->
                <v2gci_t:SalesTariffID>65</v2gci_t:SalesTariffID>
                <v2gci_t:SalesTariffDescription>SalesTariffDescription1</v2gci_t:SalesTariffDescription>
                <v2gci_t:NumEPriceLevels>3</v2gci_t:NumEPriceLevels>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>0</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>2147</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>6574</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>3</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>14937</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>24431</v2gci_t:start>
                        <v2gci_t:duration>36000</v2gci_t:duration>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                </v2gci_t:SalesTariffEntry>
            </v2gci_t:SalesTariff>
        </v2gci_t:SASchedulesTuple>
    </v2gci_t:SASchedulesList>
    <v2gci_t:AC_EVSEChargeParameter>
        <v2gci_t:AC_EVSEStatus>
            <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
            <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
            <v2gci_t:RCD>false</v2gci_t:RCD>
        </v2gci_t:AC_EVSEStatus>
        <v2gci_t:EVSENominalVoltage>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
            <v2gci_t:Unit>V</v2gci_t:Unit>
            <v2gci_t:Value>230</v2gci_t:Value>
        </v2gci_t:EVSENominalVoltage>
        <v2gci_t:EVSEMaxCurrent>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
            <v2gci_t:Unit>A</v2gci_t:Unit>
            <v2gci_t:Value>50</v2gci_t:Value>
        </v2gci_t:EVSEMaxCurrent>
```

```
                </v2gci_t:AC_EVSEChargeParameter>
            </v2gci_b:ChargeParameterDiscoveryRes>
        </v2gci_d:Body>
    </v2gci_d:V2G_Message>
```

Пример 18 сообщения V2G — Пример сообщения-ответа ChargeParameterDiscoveryRes для тарифа на основе «времени использования» с динамическим GridSchedule

J.3.5 Тариф на основе «потребления» с постоянным значением для GridSchedule

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>90813233343536637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
            <v2gci_t:SAScheduleList>
                <v2gci_t:SAScheduleTuple>
                    <v2gci_t:SAScheduleTupleID>67</v2gci_t:SAScheduleTupleID>
                    <v2gci_t:PMaxSchedule>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>86000</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>9600</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                    </v2gci_t:PMaxSchedule>
                    <v2gci_t:SalesTariff v2gci_t:id="ID001">
                        <v2gci_t:SalesTariffID>45</v2gci_t:SalesTariffID>
                        <v2gci_t:SalesTariffDescription>Standard</v2gci_t:SalesTariffDescription>
                        <v2gci_t:NumEPriceLevels>1</v2gci_t:NumEPriceLevels>
                        <v2gci_t:SalesTariffEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>86000</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                            <v2gci_t:ConsumptionCost>
                                <v2gci_t:startValue>
                                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                    <v2gci_t:Unit>W</v2gci_t:Unit>
                                    <v2gci_t:Value>0</v2gci_t:Value>
                                </v2gci_t:startValue>
                                <v2gci_t:Cost>
                                    <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                    <v2gci_t:amount>90</v2gci_t:amount>
                                    <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                                </v2gci_t:Cost>
                            </v2gci_t:ConsumptionCost>
```

```
                                <v2gci_t:ConsumptionCost>
                                    <v2gci_t:startValue>
                                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                        <v2gci_t:Unit>W</v2gci_t:Unit>
                                        <v2gci_t:Value>10000</v2gci_t:Value>
                                    </v2gci_t:startValue>
                                    <v2gci_t:Cost>
                                        <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                        <v2gci_t:amount>95</v2gci_t:amount>
                                        <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                                    </v2gci_t:Cost>
                                </v2gci_t:ConsumptionCost>
                                <v2gci_t:ConsumptionCost>
                                    <v2gci_t:startValue>
                                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                        <v2gci_t:Unit>W</v2gci_t:Unit>
                                        <v2gci_t:Value>20000</v2gci_t:Value>
                                    </v2gci_t:startValue>
                                    <v2gci_t:Cost>
                                        <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                        <v2gci_t:amount>100</v2gci_t:amount>
                                        <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                                    </v2gci_t:Cost>
                                </v2gci_t:ConsumptionCost>
                            </v2gci_t:SalesTariffEntry>
                        </v2gci_t:SalesTariff>
                    </v2gci_t:SAScheduleTuple>
                </v2gci_t:SAScheduleList>
                <v2gci_t:AC_EVSEChargeParameter>
                    <v2gci_t:AC_EVSEStatus>
                        <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
                        <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
                        <v2gci_t:RCD>false</v2gci_t:RCD>
                    </v2gci_t:AC_EVSEStatus>
                    <v2gci_t:EVSENominalVoltage>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>V</v2gci_t:Unit>
                        <v2gci_t:Value>230</v2gci_t:Value>
                    </v2gci_t:EVSENominalVoltage>
                    <v2gci_t:EVSEMaxCurrent>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>A</v2gci_t:Unit>
                        <v2gci_t:Value>50</v2gci_t:Value>
                    </v2gci_t:EVSEMaxCurrent>
                </v2gci_t:AC_EVSEChargeParameter>
            </v2gci_t:ChargeParameterDiscoveryRes>
        </v2gci_d:Body>
    </v2gci_d:V2G_Message>
```

Пример 19 сообщения V2G — Пример сообщения-ответа ChargeParameterDiscoveryRes для тарифа *на основе «потребления» с постоянным значением для GridSchedule*

J.3.6 Несколько тарифов с разными пределами запроса в GridSchedule

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
```

```
        <v2gci_h:SessionID>9091823334353837</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
            <v2gci_t:SAScheduleList>
                <v2gci_t:SAScheduleTuple>
                    <v2gci_t:SAScheduleTupleID>66</v2gci_t:SAScheduleTupleID>
                    <v2gci_t:PMaxSchedule>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>3600</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>22000</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                    </v2gci_t:PMaxSchedule>
                    <v2gci_t:SalesTariff v2gci_t:id="ID001">
                        <!-- used by XML Signature Framework for signature assignment in msg header -->
                        <v2gci_t:SalesTariffID>67</v2gci_t:SalesTariffID>
                        <v2gci_t:SalesTariffDescription>SalesTariffDescription1</v2gci_t:SalesTariffDescription>
                        <v2gci_t:NumEPriceLevels>2</v2gci_t:NumEPriceLevels>
                        <v2gci_t:SalesTariffEntry>

                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>3600</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                        </v2gci_t:SalesTariffEntry>
                    </v2gci_t:SalesTariff>
                </v2gci_t:SAScheduleTuple>
                <v2gci_t:SAScheduleTuple>
                    <v2gci_t:SAScheduleTupleID>66</v2gci_t:SAScheduleTupleID>
                    <v2gci_t:PMaxSchedule>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>3600</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>V</v2gci_t:Unit>
                                <v2gci_t:Value>11000</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                    </v2gci_t:PMaxSchedule>
                    <v2gci_t:SalesTariff v2gci_t:id="ID002">
                        <!-- used by XML Signature Framework for signature assignment in msg header -->
                        <v2gci_t:SalesTariffID>68</v2gci_t:SalesTariffID>
                        <v2gci_t:SalesTariffDescription>SalesTariffDescription2</v2gci_t:SalesTariffDescription>
                        <v2gci_t:NumEPriceLevels>2</v2gci_t:NumEPriceLevels>
                        <v2gci_t:SalesTariffEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
```

```
                                    <v2gci_t:duration>3600</v2gci_t:duration>
                                </v2gci_t:RelativeTimeInterval>
                                <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                            </v2gci_t:SalesTariffEntry>
                        </v2gci_t:SalesTariff>
                    </v2gci_t:SAScheduleTuple>
                </v2gci_t:SAScheduleList>
                <v2gci_t:AC_EVSEChargeParameter>
                    <v2gci_t:AC_EVSEStatus>
                        <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
                        <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
                        <v2gci_t:RCD>false</v2gci_t:RCD>
                    </v2gci_t:AC_EVSEStatus>
                    <v2gci_t:EVSENominalVoltage>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>V</v2gci_t:Unit>
                        <v2gci_t:Value>230</v2gci_t:Value>
                    </v2gci_t:EVSENominalVoltage>
                    <v2gci_t:EVSEMaxCurrent>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>A</v2gci_t:Unit>
                        <v2gci_t:Value>50</v2gci_t:Value>
                    </v2gci_t:EVSEMaxCurrent>
                </v2gci_t:AC_EVSEChargeParameter>
            </v2gci_p:ChargeParameterDiscoveryRes>
        </v2gci_d:Body>
    </v2gci_d:V2G_Message>
```

Пример 20 сообщения V2G — Пример сообщения-ответа ChargeParameterDiscoveryRes для нескольких тарифов с разными пределами запроса в GridSchedule

J.3.7 Тарифы на основе «времени использования» с relativePricePercentage

Пример 1 — Тариф T1 на основе «времени использования» с выбираемыми значениями ConsumptionCosts величины relativePricePercentage вида стоимости costKind.

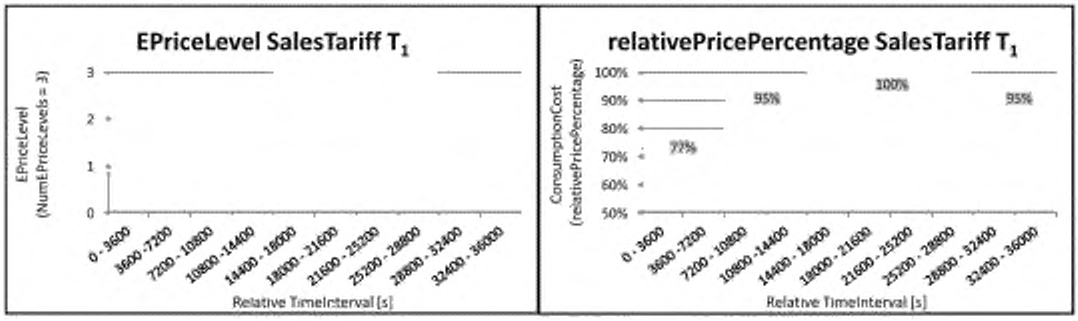


Рисунок J.1 — Примеры тарифа 1 для EPriceLevel и relativePricePercentage

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iec:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iec:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iec:15118:2:2013:MsgDataTypes"
xmlns:xmldig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iec:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9031823334353637</v2gci_h:SessionID>
    </v2gci_d:Header>
```

```
<v2gci_d:Body>
    <v2gci_b:ChargeParameterDiscoveryRes>
        <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
        <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
        <v2gci_t:SAScheduleList>
            <v2gci_t:SAScheduleTuple>
                <v2gci_t:SAScheduleTupleID>65</v2gci_t:SAScheduleTupleID>
                <v2gci_t:PMaxSchedule>
                    <v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                            <v2gci_t:duration>86000</v2gci_t:duration>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:PMax>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>22000</v2gci_t:Value>
                        </v2gci_t:PMax>
                    </v2gci_t:PMaxScheduleEntry>
                </v2gci_t:PMaxSchedule>
                <v2gci_t:SalesTariff v2gci_t:id="ID001">
                    <!-- used by XML Signature Framework for signature assignment in msg header -->
                    <v2gci_t:SalesTariffID>67</v2gci_t:SalesTariffID>
                    <v2gci_t:SalesTariffDescription>SalesTariff T1</v2gci_t:SalesTariffDescription>
                    <v2gci_t:NumEPriceLevels>3</v2gci_t:NumEPriceLevels>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>77</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>7200</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>95</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
                    <v2gci_t:SalesTariffEntry>
```

```
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>14400</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>3</v2gci_t:EPriceLevel>
                    <v2gci_t:ConsumptionCost>
                        <v2gci_t:startValue>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>0</v2gci_t:Value>
                        </v2gci_t:startValue>
                        <v2gci_t:Cost>
                            <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                            <v2gci_t:amount>100</v2gci_t:amount>
                            <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                        </v2gci_t:Cost>
                    </v2gci_t:ConsumptionCost>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>28800</v2gci_t:start>
                        <v2gci_t:duration>7200</v2gci_t:duration>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                    <v2gci_t:ConsumptionCost>
                        <v2gci_t:startValue>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>0</v2gci_t:Value>
                        </v2gci_t:startValue>
                        <v2gci_t:Cost>
                            <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                            <v2gci_t:amount>95</v2gci_t:amount>
                            <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                        </v2gci_t:Cost>
                    </v2gci_t:ConsumptionCost>
                </v2gci_t:SalesTariffEntry>
            </v2gci_t:SalesTariff>
        </v2gci_t:SAScheduleTuple>
    </v2gci_t:SAScheduleList>
    <v2gci_t:AC_EVSEChargeParameter>
        <v2gci_t:AC_EVSEStatus>
            <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
            <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
            <v2gci_t:RCD>false</v2gci_t:RCD>
        </v2gci_t:AC_EVSEStatus>
        <v2gci_t:EVSENominalVoltage>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
            <v2gci_t:Unit>V</v2gci_t:Unit>
            <v2gci_t:Value>230</v2gci_t:Value>
        </v2gci_t:EVSENominalVoltage>
        <v2gci_t:EVSEMaxCurrent>
            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
            <v2gci_t:Unit>A</v2gci_t:Unit>
            <v2gci_t:Value>60</v2gci_t:Value>
        </v2gci_t:EVSEMaxCurrent>
    </v2gci_t:AC_EVSEChargeParameter>
</v2gci_b:ChargeParameterDiscoveryRes>
</v2gci_d:Body>
</v2gci_d:V2G_Message>
```


Пример 21 сообщения V2G — Пример сообщения-ответа Charge Parameter Discovery Response
для тарифа Т1 на основе «времени использования» с relativePricePercentage

Пример 2 — Тариф T2 на основе «времени использования» с выбираемыми значениями ConsumptionCosts величины relativePricePercentage вида стоимости costKind

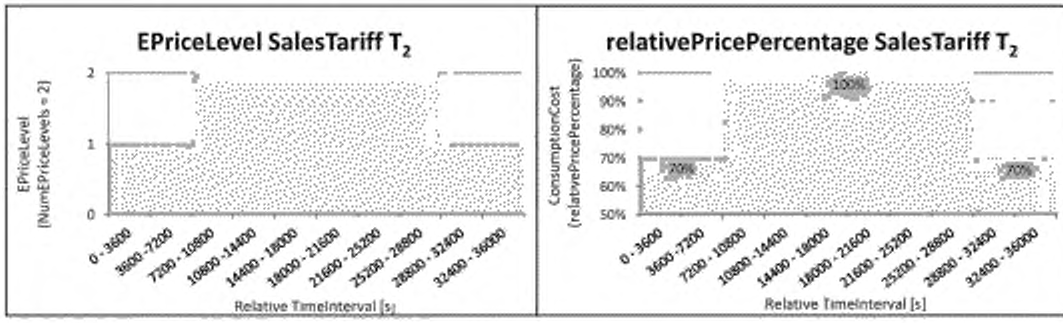


Рисунок J.2 — Примеры тарифа 2 для EPriceLevel и relativePricePercentage

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://gost.e-ecolog.ru                    ">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9031323334353637</v2gci_h:SessionID>
    </v2gci_d:Header>
    <v2gci_d:Body>
        <v2gci_b:ChargeParameterDiscoveryRes>
            <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
            <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
            <v2gci_t:SAScheduleList>
                <v2gci_t:SAScheduleTuple>
                    <v2gci_t:SAScheduleTupleID>55</v2gci_t:SAScheduleTupleID>
                    <v2gci_t:PMaxSchedule>
                        <v2gci_t:PMaxScheduleEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                                <v2gci_t:duration>36000</v2gci_t:duration>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:PMax>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>22000</v2gci_t:Value>
                            </v2gci_t:PMax>
                        </v2gci_t:PMaxScheduleEntry>
                    </v2gci_t:PMaxSchedule>
                    <v2gci_t:SalesTariff v2gci_t:id="ID001">
                        <!-- used by XML Signature Framework for signature assignment in msg header -->
                        <v2gci_t:SalesTariffID>67</v2gci_t:SalesTariffID>
                        <v2gci_t:SalesTariffDescription>SalesTariff 2</v2gci_t:SalesTariffDescription>
                        <v2gci_t:NumEPriceLevels>2</v2gci_t:NumEPriceLevels>
                        <v2gci_t:SalesTariffEntry>
                            <v2gci_t:RelativeTimeInterval>
                                <v2gci_t:start>0</v2gci_t:start>
                            </v2gci_t:RelativeTimeInterval>
                            <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
```

```
                    <v2gci_t:ConsumptionCost>
                        <v2gci_t:startValue>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>0</v2gci_t:Value>
                        </v2gci_t:startValue>
                        <v2gci_t:Cost>
                            <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                            <v2gci_t:amount>70</v2gci_t:amount>
                            <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                        </v2gci_t:Cost>
                    </v2gci_t:ConsumptionCost>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>7200</v2gci_t:start>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                    <v2gci_t:ConsumptionCost>
                        <v2gci_t:startValue>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>0</v2gci_t:Value>
                        </v2gci_t:startValue>
                        <v2gci_t:Cost>
                            <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                            <v2gci_t:amount>100</v2gci_t:amount>
                            <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                        </v2gci_t:Cost>
                    </v2gci_t:ConsumptionCost>
                </v2gci_t:SalesTariffEntry>
                <v2gci_t:SalesTariffEntry>
                    <v2gci_t:RelativeTimeInterval>
                        <v2gci_t:start>28000</v2gci_t:start>
                        <v2gci_t:duration>7200</v2gci_t:duration>
                    </v2gci_t:RelativeTimeInterval>
                    <v2gci_t:EPriceLevel>1</v2gci_t:EPriceLevel>
                    <v2gci_t:ConsumptionCost>
                        <v2gci_t:startValue>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>0</v2gci_t:Value>
                        </v2gci_t:startValue>
                        <v2gci_t:Cost>
                            <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                            <v2gci_t:amount>70</v2gci_t:amount>
                            <v2gci_t:amountMultiplier>0</v2gci_t:amountMultiplier>
                        </v2gci_t:Cost>
                    </v2gci_t:ConsumptionCost>
                </v2gci_t:SalesTariffEntry>
            </v2gci_t:SalesTariff>
        </v2gci_t:SASchedulesTuple>
    </v2gci_t:SASchedulesList>
    <v2gci_t:AC_EVSEChargeParameter>
        <v2gci_t:AC_EVSEStatus>
            <v2gci_t:NotificationMaxDelay>0</v2gci_t:NotificationMaxDelay>
            <v2gci_t:EVSENotification>None</v2gci_t:EVSENotification>
            <v2gci_t:RCD>false</v2gci_t:RCD>
        </v2gci_t:AC_EVSEStatus>
        <v2gci_t:EVSENominalVoltage>
```

```
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>V</v2gci_t:Unit>
                    <v2gci_t:Value>230</v2gci_t:Value>
                </v2gci_t:EVSENominalVoltage>
                <v2gci_t:EVSEMaxCurrent>
                    <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                    <v2gci_t:Unit>A</v2gci_t:Unit>
                    <v2gci_t:Value>60</v2gci_t:Value>
                </v2gci_t:EVSEMaxCurrent>
            </v2gci_t:AC_EVSEChargeParameter>
        </v2gci_b:ChargeParameterDiscoveryRes>
    </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример 22 сообщения V2G — Пример сообщения-ответа Charge Parameter Discovery Response тарифа T2 на основе «времени использования» с relativePricePercentage

Пример 3 — В этом примере тарифы T1 и T2 из предыдущих двух примеров представлены как два тарифа в одной схеме запрос/ответ ChargeParameterDiscovery. Следует обратить внимание на назначение других EPriceLevels для T1 и T2, а также relativePricePercentage для T1.

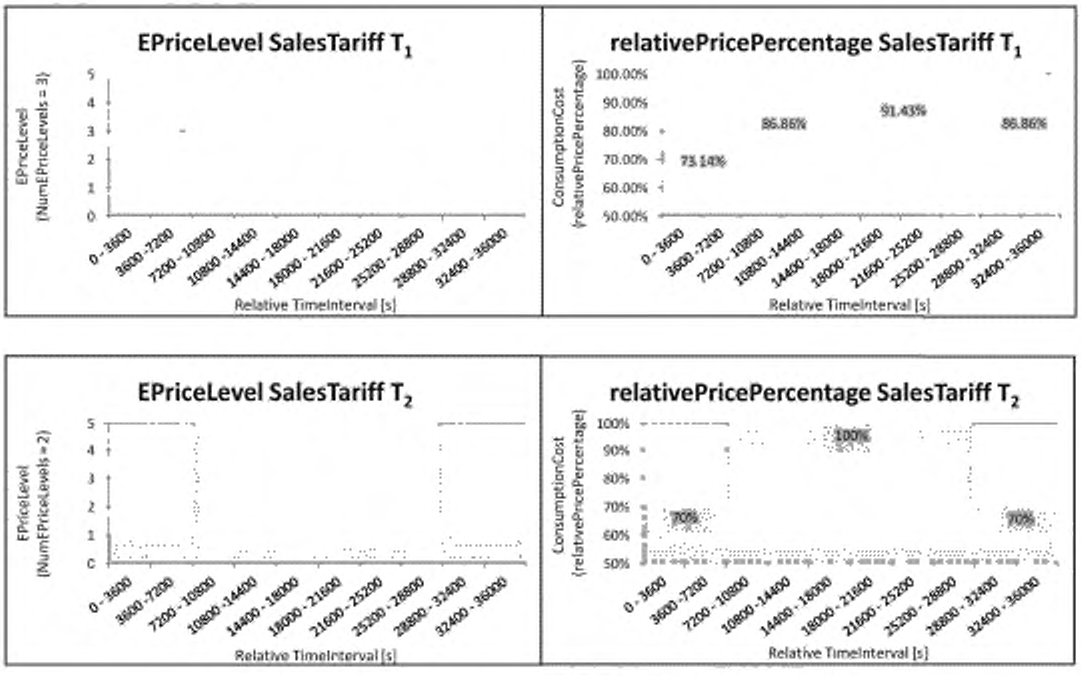


Рисунок J.3 — Примеры тарифа 3 для EPriceLevel и relativePricePercentage

```
<?xml version="1.0" encoding="UTF-8"?>
<v2gci_d:V2G_Message xmlns:v2gci_h="urn:iso:15118:2:2013:MsgHeader"
xmlns:v2gci_d="urn:iso:15118:2:2013:MsgDef"
xmlns:v2gci_t="urn:iso:15118:2:2013:MsgDataTypes"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:v2gci_b="urn:iso:15118:2:2013:MsgBody"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <v2gci_d:Header>
        <v2gci_h:SessionID>9081323934353837</v2gci_h:SessionID>
    </v2gci_d:Header>
```

```
<v2gci_d:Body>
    <v2gci_b:ChargeParameterDiscoveryRes>
        <v2gci_b:ResponseCode>OK</v2gci_b:ResponseCode>
        <v2gci_b:EVSEProcessing>Finished</v2gci_b:EVSEProcessing>
        <v2gci_t:SAScheduleList>
            <v2gci_t:SAScheduleTuple>
                <v2gci_t:SAScheduleTupleID>55</v2gci_t:SAScheduleTupleID>
                <v2gci_t:PMaxSchedule>
                    <v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                            <v2gci_t:duration>86000</v2gci_t:duration>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:PMax>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>22000</v2gci_t:Value>
                        </v2gci_t:PMax>
                    </v2gci_t:PMaxScheduleEntry>
                </v2gci_t:PMaxSchedule>
                <v2gci_t:SalesTariff v2gci_t:id="ID001">
                    <!-- used by XML Signature Framework for signature assignment in msg header -->
                    <v2gci_t:SalesTariffID>87</v2gci_t:SalesTariffID>
                    <v2gci_t:SalesTariffDescription>SalesTariff T1</v2gci_t:SalesTariffDescription>
                    <v2gci_t:NumEPriceLevels>5</v2gci_t:NumEPriceLevels>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>2</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>7314</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>-2</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>7200</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>3</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>8686</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>-2</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
```

```
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>14400</v2gci_t:start>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>4</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>9143</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>-2</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>25000</v2gci_t:start>
                            <v2gci_t:duration>7200</v2gci_t:duration>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:EPriceLevel>0</v2gci_t:EPriceLevel>
                        <v2gci_t:ConsumptionCost>
                            <v2gci_t:startValue>
                                <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                                <v2gci_t:Unit>W</v2gci_t:Unit>
                                <v2gci_t:Value>0</v2gci_t:Value>
                            </v2gci_t:startValue>
                            <v2gci_t:Cost>
                                <v2gci_t:costKind>relativePricePercentage</v2gci_t:costKind>
                                <v2gci_t:amount>6666</v2gci_t:amount>
                                <v2gci_t:amountMultiplier>-2</v2gci_t:amountMultiplier>
                            </v2gci_t:Cost>
                        </v2gci_t:ConsumptionCost>
                    </v2gci_t:SalesTariffEntry>
                </v2gci_t:SalesTariff>
            </v2gci_t:SAScheduleTuple>
            <v2gci_t:SAScheduleTuple>
                <v2gci_t:SAScheduleTupleID>64</v2gci_t:SAScheduleTupleID>
                <v2gci_t:PMaxSchedule>
                    <v2gci_t:PMaxScheduleEntry>
                        <v2gci_t:RelativeTimeInterval>
                            <v2gci_t:start>0</v2gci_t:start>
                            <v2gci_t:duration>86000</v2gci_t:duration>
                        </v2gci_t:RelativeTimeInterval>
                        <v2gci_t:PMax>
                            <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                            <v2gci_t:Unit>W</v2gci_t:Unit>
                            <v2gci_t:Value>22000</v2gci_t:Value>
                        </v2gci_t:PMax>
                    </v2gci_t:PMaxScheduleEntry>
                </v2gci_t:PMaxSchedule>
                <v2gci_t:SalesTariff v2gci_t:id="ID002">
                    <!-- used by XML Signature Framework for signature assignment in msg header -->
                    <v2gci_t:SalesTariffID>66</v2gci_t:SalesTariffID>
                    <v2gci_t:SalesTariffDescription>SalesTariff 2</v2gci_t:SalesTariffDescription>
                    <v2gci_t:NumEPriceLevels>5</v2gci_t:NumEPriceLevels>
                    <v2gci_t:SalesTariffEntry>
                        <v2gci_t:RelativeTimeInterval>
```

```
                    <v2gci_tstart>0</v2gci_tstart>
                </v2gci_tRelativeTimeInterval>
                <v2gci_tEPriceLevel>1</v2gci_tEPriceLevel>
                <v2gci_tConsumptionCost>
                    <v2gci_tstartValue>
                        <v2gci_tMultiplier>0</v2gci_tMultiplier>
                        <v2gci_tUnit>W</v2gci_tUnit>
                        <v2gci_tValue>0</v2gci_tValue>
                    </v2gci_tstartValue>
                    <v2gci_tCost>
                        <v2gci_tcostKind>relativePricePercentage</v2gci_tcostKind>
                        <v2gci_tamount>70</v2gci_tamount>
                        <v2gci_tamountMultiplier>0</v2gci_tamountMultiplier>
                    </v2gci_tCost>
                </v2gci_tConsumptionCost>
            </v2gci_tSalesTariffEntry>
            <v2gci_tSalesTariffEntry>
                <v2gci_tRelativeTimeInterval>
                    <v2gci_tstart>7200</v2gci_tstart>
                </v2gci_tRelativeTimeInterval>
                <v2gci_tEPriceLevel>5</v2gci_tEPriceLevel>
                <v2gci_tConsumptionCost>
                    <v2gci_tstartValue>
                        <v2gci_tMultiplier>0</v2gci_tMultiplier>
                        <v2gci_tUnit>W</v2gci_tUnit>
                        <v2gci_tValue>0</v2gci_tValue>
                    </v2gci_tstartValue>
                    <v2gci_tCost>
                        <v2gci_tcostKind>relativePricePercentage</v2gci_tcostKind>
                        <v2gci_tamount>100</v2gci_tamount>
                        <v2gci_tamountMultiplier>0</v2gci_tamountMultiplier>
                    </v2gci_tCost>
                </v2gci_tConsumptionCost>
            </v2gci_tSalesTariffEntry>
            <v2gci_tSalesTariffEntry>
                <v2gci_tRelativeTimeInterval>
                    <v2gci_tstart>28000</v2gci_tstart>
                    <v2gci_tduration>7200</v2gci_tduration>
                </v2gci_tRelativeTimeInterval>
                <v2gci_tEPriceLevel>1</v2gci_tEPriceLevel>
                <v2gci_tConsumptionCost>
                    <v2gci_tstartValue>
                        <v2gci_tMultiplier>0</v2gci_tMultiplier>
                        <v2gci_tUnit>W</v2gci_tUnit>
                        <v2gci_tValue>0</v2gci_tValue>
                    </v2gci_tstartValue>
                    <v2gci_tCost>
                        <v2gci_tcostKind>relativePricePercentage</v2gci_tcostKind>
                        <v2gci_tamount>70</v2gci_tamount>
                        <v2gci_tamountMultiplier>0</v2gci_tamountMultiplier>
                    </v2gci_tCost>
                </v2gci_tConsumptionCost>
            </v2gci_tSalesTariffEntry>
        </v2gci_tSalesTariff>
    </v2gci_tSAScheduleTuple>
</v2gci_tSAScheduleList>
<v2gci_tAC_EVSEChargeParameter>
    <v2gci_tAC_EVSEStatus>
        <v2gci_tNotificationMaxDelay>0</v2gci_tNotificationMaxDelay>
        <v2gci_tEVSENotification>None</v2gci_tEVSENotification>
        <v2gci_tRCD>false</v2gci_tRCD>
```

```
                    </v2gci_t:AC_EVSEStatus>
                    <v2gci_t:EVSENominalVoltage>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>V</v2gci_t:Unit>
                        <v2gci_t:Value>230</v2gci_t:Value>
                    </v2gci_t:EVSENominalVoltage>
                    <v2gci_t:EVSEMaxCurrent>
                        <v2gci_t:Multiplier>0</v2gci_t:Multiplier>
                        <v2gci_t:Unit>A</v2gci_t:Unit>
                        <v2gci_t:Value>50</v2gci_t:Value>
                    </v2gci_t:EVSEMaxCurrent>
                </v2gci_t:AC_EVSEChargeParameter>
            </v2gci_h:ChargeParameterDiscoveryRes>
        </v2gci_d:Body>
</v2gci_d:V2G_Message>
```

Пример Z3 сообщения V2G — Пример сообщения-ответа Charge Parameter Discovery Response для тарифов T1 и T2 на основе «времени использования» с relativePricePercentage

Приложение К
(справочное)

Привязка элементов случаев использования части 1

К.1 Отношение идентификационных режимов и элементов случаев использования

Идентификационные режимы внешних средств идентификации (EIM) и «Plug and Charge» (PnC) охватывают разные элементы случаев использования, описанные в части 1. В таблице К.1 дан подробный обзор последовательностей сообщений, описанных в подразделе 8.6, и охватываемых элементов А1—Н1 случаев использования, описанных в части 1.

Таблица К.1 — Наборы сообщений и охватываемые элементы случаев использования

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: VAS
supportedAppProtocolReq	B1	B1	B1	B1	–	–	–
ProtocolNamespace	B1	B1	B1	B1	–	–	–
VersionNumberMajor	B1	B1	B1	B1	–	–	–
VersionNumberMinor	B1	B1	B1	B1	–	–	–
SchemaID	B1	B1	B1	B1	–	–	–
Priority	B1	B1	B1	B1	–	–	–
supportedAppProtocolRes	B1	B1	B1	B1	–	–	–
ResponseCode	B1	B1	B1	B1	–	–	–
SchemaID	B1	B1	B1	B1	–	–	–
SessionSetupReq	B1	B1	B1	B1	–	–	–
Header	B1	B1	B1	B1	–	–	–
SessionId	B1	B1	B1	B1	–	–	–
Notification	B1	B1	B1	B1	–	–	–
FaultCode	B1	B1	B1	B1	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPT
FaultMsg	B1	B1	B1	B1	–	–	–
Signature (см. отдельную таблицу)	B1	B1	B1	B1	–	–	–
Body	B1	B1	B1	B1	–	–	–
EVCCID	B1	B1	B1	B1	–	–	–
SessionSetupRes	B1	B1	B1	B1	–	–	–
Header	B1	B1	B1	B1	–	–	–
SessionId	B1	B1	B1	B1	–	–	–
Notification	B1	B1	B1	B1	–	–	–
FaultCode	B1	B1	B1	B1	–	–	–
FaultMsg	B1	B1	B1	B1	–	–	–
Signature (см. отдельную таблицу)	–	–	B1	B1	–	–	–
Body	B1	B1	B1	B1	–	–	–
ResponseCode	B1	B1	B1	B1	–	–	–
EVSEID	B1	B1	B1	B1	–	–	–
DateTimeNow	B1	B1	B1	B1	–	–	–
ServiceDiscoveryReq	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
Header	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
SessionId	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
Notification	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–

Продолжение таблицы К.1

	Зарезерв. пространство токена с ЕЦИ	Зарезерв. пространство токена с РНС	Зарезерв. пространство токена с РНС	Опция: Обязательное подтверждение	Опция: Устаревшее подтверждение	Опция: WAS
FaultCode	D3, D4	D1, D2	D1, D2	–	–	–
FaultMsg	D3, D4	D1, D2	D1, D2	–	–	–
Signature	D3,	D1.	D1.	–	–	–
	D4	D2	D2			
Body	D3, D4	D1, D2	D1, D2	–	–	–
ServiceScope	D3, D4	D1, D2	D1, D2	–	–	–
ServiceCategory	D3, D4	D1, D2	D1, D2	–	–	–
ServiceDiscoveryRes	D3, D4	D1, D2	D1, D2	–	–	–
Header	D3, D4	D1, D2	D1, D2	–	–	–
SessionId	D3, D4	D1, D2	D1, D2	–	–	–
Notification	D3, D4	D1, D2	D1, D2	–	–	–
FaultCode	D3, D4	D1, D2	D1, D2	–	–	–
FaultMsg	D3, D4	D1, D2	D1, D2	–	–	–
Signature	–	D1, D2	D1, D2	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с ЕВ	Зарядка постоянным током с ЕВ	Зарядка переменным током с РЗС	Зарядка постоянным током с РЗС	Опция: Обязательно поддерживается	Опция: Устаревшее поддерживается	Опция: WAS
Body	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ResponseCode	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
PaymentOptions	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
PaymentOption	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
PaymentOption	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ChargeService	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ServiceID	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ServiceName	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ServiceCategory	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ServiceScope	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
FreeService	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
SupportedEnergyTransferMode	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
EnergyTransferMode	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–

Продолжение таблицы К.1

	Запрещен переопределен только с EIM	Запрещен постоянный только с EIM	Запрещен переопределен только с FHC	Запрещен постоянный только с FHC	Опция: Обязательно поддерживается	Опция: Желательно поддерживается	Опция: WAS
ServiceList	–	–	–	–	C1	C2	G1, G2
Service	–	–	–	–	C1	C2	G1, G2
ServiceID	–	–	–	–	C1	C2	G1, G2
ServiceName	–	–	–	–	C1	C2	G1, G2
ServiceCategory	–	–	–	–	C1	C2	G1, G2
ServiceScope	–	–	–	–	C1	C2	G1, G2
FreeService	–	–	–	–	C1	C2	G1, G2
ServiceDetailReq	–	–	–	–	C1	C2	G1, G2
Header	–	–	–	–	C1	C2	G1, G2
SessionId	–	–	–	–	C1	C2	G1, G2
Notification	–	–	–	–	C1	C2	G1, G2
FaultCode	–	–	–	–	C1	C2	G1, G2
FaultMsg	–	–	–	–	C1	C2	G1, G2

Продолжение таблицы К.1

	Зарезервировано только с ЕМ	Зарезервировано только с ЕМ	Зарезервировано только с РКС	Зарезервировано только с РКС	Опция: Обязательно поддерживается	Опция: Желательно поддерживается	Опция: WAS
Signature	–	–	–	–	C1	C2	G1, G2
Body	–	–	–	–	C1	C2	G1, G2
ServiceID	–	–	–	–	C1	C2	G1, G2
ServiceDetailRes	–	–	–	–	C1	C2	G1, G2
Header	–	–	–	–	C1	C2	G1, G2
SessionId	–	–	–	–	C1	C2	G1, G2
Notification	–	–	–	–	C1	C2	G1, G2
FaultCode	–	–	–	–	C1	C2	G1, G2
FaultMsg	–	–	–	–	C1	C2	G1, G2
Signature	–	–	–	–	C1	C2	G1, G2
Body	–	–	–	–	C1	C2	G1, G2
ResponseCode	–	–	–	–	C1	C2	G1, G2
ServiceID	–	–	–	–	C1	C2	G1, G2

Продолжение таблицы К.1

	Вариант применения только с ЕМ	Вариант постоянная только с РКС	Вариант применения только с РКС	Вариант постоянная только с ЕМ	Вариант: Обязательно подтверждается	Вариант: Желательно подтверждается	Вариант: WAS
ServiceParameterList	–	–	–	–	C1	C2	G1, G2
Parameter Set	–	–	–	–	C1	C2	G1, G2
ParameterSet ID	–	–	–	–	C1	C2	G1, G2
Parameter	–	–	–	–	C1	C2	G1, G2
ServicePaymentSelectionReq	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
Header	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
SessionId	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
Notification	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
FaultCode	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
FaultMsg	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
Signature (см. отдельную таблицу)	–	D1, D2	D1, D2	–	C1	C2	G1, G2
Body	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2
SelectedPaymentOption	D3, D4	D1, D2	D1, D2	D3, D4	C1	C2	G1, G2

Продолжение таблицы К.1

	Зарядная передвижная только с ЕМ	Зарядная постоянная только с ЕМ	Зарядная передвижная только с РКС	Зарядная постоянная только с РКС	Опция: Обязательно поддерживается	Опция: Условно поддерживается	Опция: WAS
SelectedServiceList	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
SelectedService	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
ServiceID	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
ParameterSet ID	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
ServicePaymentSelectionReq	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
Header	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
SessionId	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
Notification	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
FaultCode	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
FaultMsg	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
Signature (см. отдельную таблицу)	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
Body	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2
ResponseCode	D3, D4	D3, D4	D1, D2	D1, D2	C1	C2	G1, G2

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WS
PaymentDetailsReq	–	–	D1, D2	D1, D2	–	–	–
Header	–	–	D1, D2	D1, D2	–	–	–
SessionId	–	–	D1, D2	D1, D2	–	–	–
Notification	–	–	D1, D2	D1, D2	–	–	–
FaultCode	–	–	D1, D2	D1, D2	–	–	–
FaultMsg	–	–	D1, D2	D1, D2	–	–	–
Signature (см. отдельную таблицу)	–	–	D1, D2	D1, D2	–	–	–
Body	–	–	D1, D2	D1, D2	–	–	–
eMAID	–	–	D1, D2	D1, D2	–	–	–
ContractSignatureCertChain	–	–	D1, D2	D1, D2	–	–	–
Certificate	–	–	D1, D2	D1, D2	–	–	–
SubCertificates	–	–	D1, D2	D1, D2	–	–	–
Certificate	–	–	D1, D2	D1, D2	–	–	–

Продолжение таблицы К.1

	Зарплата переключена только с ЕЦК	Зарплата поступила только с ЕЦК	Зарплата переключена только с РЦС	Зарплата поступила только с РЦС	Отдел: Обновление подтверждается	Отдел: Установка подтверждается	Отдел: ИС
PaymentDetailsRes	–	–	D1, D2	D1, D2	–	–	–
Header	–	–	D1, D2	D1, D2	–	–	–
SessionId	–	–	D1, D2	D1, D2	–	–	–
Notification	–	–	D1, D2	D1, D2	–	–	–
FaultCode	–	–	D1, D2	D1, D2	–	–	–
FaultMsg	–	–	D1, D2	D1, D2	–	–	–
Signature	–	–	D1, D2	D1, D2	–	–	–
Body	–	–	D1, D2	D1, D2	–	–	–
ResponseCode	–	–	D1, D2	D1, D2	–	–	–
GenChallenge	–	–	D1, D2	D1, D2	–	–	–
DateTimeNow	–	–	D1, D2	D1, D2	–	–	–
AuthorizationReq	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
Header	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–

Продолжение таблицы К.1

	Зеркалит перекрещенная тесьма с ЕЦ	Зеркалит плотницкая тесьма с ЕЦ	Зеркалит плотницкая тесьма с РЧС	Отдых: Обязательно подтверждается	Отдых: Устаканова подтверждается	Отдых: VAS
SessionId	D3, D4	D3, D4	D1, D2	–	–	–
Notification	D3, D4	D3, D4	D1, D2	–	–	–
FaultCode	D3, D4	D3, D4	D1, D2	–	–	–
FaultMsg	D3, D4	D3, D4	D1, D2	–	–	–
Signature	D3, D4	D3, D4	D1, D2	–	–	–
Body	D3, D4	D3, D4	D1, D2	–	–	–
GenChallenge	D3, D4	D3, D4	D1, D2	–	–	–
AuthorizationRes	D3, D4	D3, D4	D1, D2	–	–	–
Header	D3, D4	D3, D4	D1, D2	–	–	–
SessionId	D3, D4	D3, D4	D1, D2	–	–	–
Notification	D3, D4	D3, D4	D1, D2	–	–	–
FaultCode	D3, D4	D3, D4	D1, D2	–	–	–
FaultMsg	D3, D4	D3, D4	D1, D2	–	–	–

Продолжение таблицы К.1

	Энергия переменная только с ЕМ	Энергия постоянная только с ЕМ	Энергия переменная только с РЕС	Энергия постоянная только с РЕС	Опция: Отключение переподключается	Опция: Остановка переподключается	Опция: VAS
Signature	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
Body	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
EVSEProcessing	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ResponseCode	D3, D4	D3, D4	D1, D2	D1, D2	–	–	–
ChargeParameterDiscoveryRes	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Header	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SessionId	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Notification	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultCode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultMsg	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–

Продолжение таблицы К.1

	Энергия переменная токов с ЕМ	Энергия постоянная токов с ЕМ	Энергия переменная токов с РНС	Энергия постоянная токов с РНС	Опция: Обязательно поддерживается	Опция: Желательно поддерживается	Опция: VAS
Signature	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Body	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
MaxEntriesSASchedule Tuple	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
RequestedEnergyTransferMode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
AC_EVChargeParameter	E1, E2	–	E3, E5, F3	–	–	–	–
DepartureTime	E1, E2	–	E3, E5, F3	–	–	–	–
Eamount	E1, E2	–	E3, E5, F3	–	–	–	–
EVMaxVoltage	E1, E2	–	E3, E5, F3	–	–	–	–
EVMaxCurrent	E1, E2	–	E3, E5, F3	–	–	–	–
EVMinCurrent	E1, E2	–	E3, E5, F3	–	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPS
DC_EVChargeParameter	–	E4	–	E4, F3	–	–	–
DepartureTime	–	E4	–	E4, F3	–	–	–
DC_EVStatus	–	E4	–	E4, F3	–	–	–
EVReady	–	E4	–	E4, F3	–	–	–
EVRESSConditioning	–	E4	–	E4, F3	–	–	–
EVErrorCode	–	E4	–	E4, F3	–	–	–
EVRESSSOC	–	E4	–	E4, F3	–	–	–
EVMaximumCurrentLimit	–	E4	–	E4, F3	–	–	–
EVSEMaximumPowerLimit	–	E4	–	E4, F3	–	–	–
EVMaximumVoltageLimit	–	E4	–	E4, F3	–	–	–
EVEnergyCapacity	–	E4	–	E4, F3	–	–	–
EVEnergyRequest	–	E4	–	E4, F3	–	–	–
FullSOC	–	E4	–	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Отдача: обязательно подтверждается	Отдача: желательно подтверждается	Отдача: VAS
Bulk2OC	–	E4	–	E4, F3	–	–	–
ChargeParameterDiscoveryReq	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Header	E1, E2	E4	E3, E6, F3	E4, F3	–	–	–
SessionId	E1, E2	E4	E3, E5,	E4, F3	–	–	–
			F3				
Notification	E1, E2	E4	E3, E6, F3	E4, F3	–	–	–
FaultCode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultMsg	E1, E2	E4	E3, E6, F3	E4, F3	–	–	–
Signature	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Body	E1, E2	E4	E3, E6, F2	E4, F3	–	–	–
ResponseCode	E1, E2	E4	E3, E6, F2	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядная передающая точка ⇔ ЕВМ	Зарядная приемная точка ⇔ ЕВМ	Зарядная передающая точка ⇔ РМС	Зарядная приемная точка ⇔ РМС	Опция: Обязательное подтверждение	Опция: Эстафетное подтверждение	Опция: WAS
EVSEProcessing	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SASchedule	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SASScheduleTuple	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SASScheduleTupleID	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
PMaxSchedule	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
PMaxScheduleID	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
PMAxScheduleEntry	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
RelativeTimeInterval	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
start	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
duration	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка периодическая только с ЕМ	Зарядка постоянная только с ЕМ	Зарядка периодическая только с РНС	Зарядка постоянная только с РНС	Опция: обязательно поддерживается	Опция: желательно поддерживается	Опция: WAS
Pmax	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SalesTariff	–	–	E3, E5, F3	E4, F3	–	–	–
SalesTariffID	–	–	E3, E5, F3	E4, F3	–	–	–
SalesTariffDescription	–	–	E3, E5, F3	E4, F3	–	–	–
NumEPriceLevels	–	–	E3, E5, F3	E4, F3	–	–	–
SalesTariffEntry	–	–	E3, E5, F3	E4, F3	–	–	–
RelativeTimeInterval	–	–	E3, E5, F3	E4, F3	–	–	–
start	–	–	E3, E5, F3	E4, F3	–	–	–
duration	–	–	E3, E5, F3	E4, F3	–	–	–
EPriceLevel	–	–	E3, E5, F3	E4, F3	–	–	–

Продолжение таблицы К.1

	Операция поставщика только с EIM	Операция поставщика только с PnC	Операция пользователя только с PnC	Опция: Обязательно поддерживается	Опция: Устаревшее поддерживается	Опция: WAS
ConsumptionCost	–	E4, F3	E3, E5, F2	–	–	–
Start Value	–	E4, F3	E3, E5, F2	–	–	–
Cost	–	E4, F3	E3, E5, F2	–	–	–
AC_EVSEChargeParameter	E1, E2	–	E3, E5, F2	–	–	–
AC_EVSEStatus	E1, E2	–	E3, E5, F2	–	–	–
RCD	E1, E2	–	E3, E5, F2	–	–	–
NotificationMaxDelay	E1, E2	–	E3, E5, F2	–	–	–
EVSENotification	E1, E2	–	E3, E5, F2	–	–	–
EVSENominalVoltage	E1, E2	–	E3, E5, F2	–	–	–
EVSEMaxCurrent	E1, E2	–	E3, E5, F2	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WS
DC_EVSEChargeParameter	–	E4	–	E4, F3	–	–	–
DC_EVSEStatus	–	E4	–	E4, F3	–	–	–
EVSEIsolationStatus	–	E4	–	E4, F3	–	–	–
EVSEStatusCode	–	E4	–	E4, F3	–	–	–
NotificationMaxDelay	–	E4	–	E4, F3	–	–	–
EVSENotification	–	E4	–	E4, F3	–	–	–
EVSEMaximumCurrentLimit	–	E4	–	E4, F3	–	–	–
EVSEMaximumPowerLimit	–	E4	–	E4, F3	–	–	–
EVSEMaximumVoltageLimit	–	E4	–	E4, F3	–	–	–
EVSEMinimumCurrentLimit	–	E4	–	E4, F3	–	–	–
EVSEMinimumVoltageLimit	–	E4	–	E4, F3	–	–	–
EVSECurrentRegulationTolerance	–	E4	–	E4, F3	–	–	–
EVSEPeakCurrentRipple	–	E4	–	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление программного обеспечения	Опция: Установка программного обеспечения	Опция: VAS
EVSEEnergyToBeDelivered	–	E4	–	E4, F3	–	–	–
PowerDeliveryReq	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Header	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SessionId	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Notification	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultCode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultMsg	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Signature (см. отдельную таблицу)	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Body	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ChargeProgress	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с ЕВК	Зарядка постоянным током с ЕВК	Зарядка переменным током с РНС	Зарядка постоянным током с РНС	Отдых: Обнаружение подтверждается	Отдых: Устранение подтверждается	Отдых: WAS
SASScheduleTupleID	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ChargingProfile	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ProfileEntry	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ChargingProfileEntryStart	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ChargingProfileEntryMaxPower	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ChargingProfileEntryMaxNumberOfPhasesInUse	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
DC_EVPowerDeliveryParameter	–	E4	–	E4, F3	–	–	–
DC_EVStatus	–	E4	–	E4, F3	–	–	–
EVReady	–	E4	–	E4, F3	–	–	–
EVRESSConditioning	–	E4	–	E4, F3	–	–	–
EVErrorCode	–	E4	–	E4, F3	–	–	–

Продолжение таблицы К.1

	Энергия переменная только с EIM	Энергия постоянная только с EIM	Энергия переменная только с PnC	Энергия постоянная только с PnC	Опция: Обязательно поддерживается	Опция: Желательно поддерживается	Опция: VAS
EVRESSSOC	–	E4	–	E4, F3	–	–	–
BulkChargingComplete	–	E4	–	E4, F3	–	–	–
ChargingComplete	–	E4	–	E4, F3	–	–	–
PowerDeliveryRes	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Header	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
SessionId	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Notification	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultCode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
FaultMsg	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
Signature (см. отдельную таблицу)	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обязательно поддерживается	Опция: Желательно поддерживается	Опция: VAS
Body	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
ResponseCode	E1, E2	E4	E3, E5, F3	E4, F3	–	–	–
AC_EVSEStatus	E1, E2	–	E3, E5, F3	–	–	–	–
RCD	E1, E2	–	E3, E5, F3	–	–	–	–
NotificationMaxDelay	E1, E2	–	E3, E5, F3	–	–	–	–
EVSENotification	E1, E2	–	E3, E5, F3	–	–	–	–
DC_EVSEStatus	–	E4	–	E4, F3	–	–	–
EVSEIsolationStatus	–	E4	–	E4, F3	–	–	–
EVSEStatusCode	–	E4	–	E4, F3	–	–	–
NotificationMaxDelay	–	E4	–	E4, F3	–	–	–
EVSENotification	–	E4	–	E4, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPT
CertificateUpdateReq	–	–	–	–	C1	–	–
Header	–	–	–	–	C1	–	–
SessionId	–	–	–	–	C1	–	–
Notification	–	–	–	–	C1	–	–
FaultCode	–	–	–	–	C1	–	–
FaultMsg	–	–	–	–	C1	–	–
Signature (см. отдельную таблицу)	–	–	–	–	C1	–	–
Body	–	–	–	–	C1	–	–
ContractSignatureCertChain	–	–	–	–	C1	–	–
Certificate	–	–	–	–	C1	–	–
SubCertificates	–	–	–	–	C1	–	–
Certificate	–	–	–	–	C1	–	–
eMAID	–	–	–	–	C1	–	–
ListOfRootCertificateIDs	–	–	–	–	C1	–	–
RootCertificateID	–	–	–	–	C1	–	–
CertificateUpdateRes	–	–	–	–	C1	–	–
Header	–	–	–	–	C1	–	–
SessionId	–	–	–	–	C1	–	–
Notification	–	–	–	–	C1	–	–
FaultCode	–	–	–	–	C1	–	–
FaultMsg	–	–	–	–	C1	–	–

Продолжение таблицы К.1

	Электронная подпись токена с ЕЦК	Электронная подпись токена с РЧС	Электронная подпись токена с РЧС	Опция: Обновление программного обеспечения	Опция: Эластичное программное обеспечение	Опция: VMS
Signature (см. отдельную таблицу)	–	–	–	C1	–	–
Body	–	–	–	C1	–	–
ResponseCode	–	–	–	C1	–	–
SAProvisioningCertificateChain				C1	–	–
ContractSignatureCertChain	–	–	–	C1	–	–
Certificate	–	–	–	C1	–	–
subCertificates	–	–	–	C1	–	–
Certificate	–	–	–	C1	–	–
ContractSignatureEncrypted PrivateKey	–	–	–	C1	–	–
DhpubKey	–	–	–	C1	–	–
eMAID	–	–	–	C1	–	–
RetryCounter	–	–	–	C1	–	–
CertificateInstallationReq	–	–	–	–	C2	–
Header	–	–	–	–	C2	–
SessionId	–	–	–	–	C2	–
Notification	–	–	–	–	C2	–
FaultCode	–	–	–	–	C2	–
FaultMsg	–	–	–	–	C2	–
Signature (см. отдельную таблицу)	–	–	–	–	C2	–
Body	–	–	–	–	C2	–
OEMProvisioningCert	–	–	–	–	C2	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление программного обеспечения	Опция: Устаревшее программное обеспечение	Опция: VAS
ListOfRootCertificateIDs	–	–	–	–	–	C2	–
RootCertificateID	–	–	–	–	–	C2	–
CertificateInstallationRes	–	–	–	–	–	C2	–
Header	–	–	–	–	–	C2	–
SessionId	–	–	–	–	–	C2	–
Notification	–	–	–	–	–	C2	–
FaultCode	–	–	–	–	–	C2	–
FaultMsg	–	–	–	–	–	C2	–
Signature (см. отдельную таблицу)	–	–	–	–	–	C2	–
Body	–	–	–	–	–	C2	–
ResponseCode	–	–	–	–	–	C2	–
SAProvisioningCertificateChain					–	C2	–
ContractSignatureCertChain	–	–	–	–	–	C2	–
Certificate	–	–	–	–	–	C2	–
SubCertificates	–	–	–	–	–	C2	–
Certificate	–	–	–	–	–	C2	–
ContractSignatureEncryptedPrivateKey	–	–	–	–	–	C2	–
DHpublickey	–	–	–	–	–	C2	–
eMAID	–	–	–	–	–	–	–
SessionStopReq	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обязательное подтверждение	Опция: Встроенное подтверждение	Опция: VAS
Header	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
SessionId	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
Notification	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
FaultCode	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
FaultMsg	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
Signature	–	–	–	–	–	–	–
Body	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
ChargingSession	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
SessionStopRes	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
Header	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
SessionId	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
Notification	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
FaultCode	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
FaultMsg	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление программного обеспечения	Опция: Установка программного обеспечения	Опция: VAS
Signature ()	–	–	H1, F3	H1, F3	–	–	–
Body	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
ResponseCode	H1, F3	H1, F3	H1, F3	H1, F3	–	–	–
ChargingStatusReq	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
Header	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
SessionId	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
Notification	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
FaultCode	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
FaultMsg	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–

Продолжение таблицы К.1

	Операция передачи токена с ЕМ	Операция подтверждения токена с ЕМ	Операция передачи токена с РПС	Операция подтверждения токена с РПС	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: VAS
Signature	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
Body	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
ChargingStatusRes	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
Header	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
SessionId	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
Notification	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
FaultCode	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
FaultMsg	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление программного обеспечения	Опция: Установка программного обеспечения	Опция: VAS
Signature	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
Body	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
ResponseCode	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
EVSEID	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
SASchedulaTupleID	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
EVSEMaxCurrent	E1, E2, F0	–	E3, E5, F0,F1	–	–	–	–
MeterInfo	E1,	–	E3,	–	–	–	–
	E2, F0		E5, F0,F1				

Продолжение таблицы К.1

	Запрещен периодический поток с ЕМ	Запрещен постоянный поток с ЕМ	Запрещен периодический поток с РЧС	Запрещен постоянный поток с РЧС	Опция: Обязательное подтверждение	Опция: Устаревшее подтверждение	Опция: WAS
MeterID	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
MeterReading	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
SigMeterReading	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
MeterStatus	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
TMeter	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
ReceiptRequired	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
AC_EVSEStatus	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–
RCD	E1, E2, F0	–	E3, E4, F0, F1	–	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPT
NotificationMaxDelay	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
EVSENotification	E1, E2, F0	–	E3, E5, F0, F1	–	–	–	–
MeteringReceiptReq	–	–	F1	F1	–	–	–
Header	–	–	F1	F1	–	–	–
SessionId	–	–	F1	F1	–	–	–
Notification	–	–	F1	F1	–	–	–
FaultCode	–	–	F1	F1	–	–	–
FaultMsg	–	–	F1	F1	–	–	–
Signature	–	–	F1	F1	–	–	–
Body	–	–	F1	F1	–	–	–
SessionID	–	–	F1	F1	–	–	–
SASchedulaTupleID	–	–	F1	F1	–	–	–
MeterInfo	–	–	F1	F1	–	–	–
MeterID	–	–	F1	F1	–	–	–
MeterReading	–	–	F1	F1	–	–	–
SigMeterReading	–	–	F1	F1	–	–	–
MeterStatus	–	–	F1	F1	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с ЕІМ	Зарядка постоянным током с ЕІМ	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WS
Tmeter	–	–	F1	F1	–	–	–
MeteringReceiptRes	–	–	F1	F1	–	–	–
Header	–	–	F1	F1	–	–	–
SessionId	–	–	F1	F1	–	–	–
Notification	–	–	F1	F1	–	–	–
FaultCode	–	–	F1	F1	–	–	–
FaultMsg	–	–	F1	F1	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–
Body	–	–	F1	F1	–	–	–
AC_EVSEStatus	–	–	F1	F1	–	–	–
RCD	–	–	F1	F1	–	–	–
NotificationMaxDelay	–	–	F1	F1	–	–	–
EVSENotification	–	–	F1	F1	–	–	–
DC_EVSEStatus	–	–	F1	F1	–	–	–
EVSEIsolationStatus	–	–	F1	F1	–	–	–
EVSEStatusCode	–	–	F1	F1	–	–	–
NotificationMaxDelay	–	–	F1	F1	–	–	–
EVSENotification	–	–	F1	F1	–	–	–
ResponseCode	–	–	F1	F1	–	–	–
CableCheckReq	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPS
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	E4	–	E4	–	–	–
Body	–	E4	–	E4	–	–	–
DC_EVStatus	–	E4	–	E4	–	–	–
EVReady	–	E4	–	E4	–	–	–
EVRESSConditioning	–	E4	–	E4	–	–	–
EVErrorCode	–	E4	–	E4	–	–	–
EVRESSSOC	–	E4	–	E4	–	–	–
CableCheckRes	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	E4	–	E4	–	–	–
Body	–	E4	–	E4	–	–	–
ResponseCode	–	E4	–	E4	–	–	–
EVSEProcessing	–	E4	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядная переговорная точка с EIM	Зарядная постоянная точка с EIM	Зарядная переговорная точка с PnC	Зарядная постоянная точка с PnC	Опция: Обязательно поддерживается	Опция: Устаревшее поддерживается	Опция: VAS
DC_EVSEStatus	–	E4	–	E4	–	–	–
EVSEIsolationStatus	–	E4	–	E4	–	–	–
EVSEStatusCode	–	E4	–	E4	–	–	–
NotificationMaxDelay	–	E4	–	E4	–	–	–
EVSENotification	–	E4	–	E4	–	–	–
PreChargeReq	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature	–	E4	–	E4	–	–	–
Body	–	E4	–	E4	–	–	–
DC_EVStatus	–	E4	–	E4	–	–	–
EVReady	–	E4	–	E4	–	–	–
EVRESSConditioning	–	E4	–	E4	–	–	–
EVErrorCode	–	E4	–	E4	–	–	–
EVRESSSOC	–	E4	–	E4	–	–	–
EVTargetVoltage	–	E4	–	E4	–	–	–
EVTargetCurrent	–	E4	–	E4	–	–	–
PreChargeRes	–	E4	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPS
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	E4	–	E4	–	–	–
Body	–	E4	–	E4	–	–	–
ResponseCode	–	E4	–	E4	–	–	–
DC_EVSEStatus	–	E4	–	E4	–	–	–
EVSEIsolationStatus	–	E4	–	E4	–	–	–
EVSEStatusCode	–	E4	–	E4	–	–	–
NotificationMaxDelay	–	E4	–	E4	–	–	–
EVSENotification	–	E4	–	E4	–	–	–
EVSEPresentVoltage	–	E4	–	E4	–	–	–
CurrentDemandReq	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	E4	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WS
Body	–	E4	–	E4	–	–	–
DC_EVStatus	–	E4	–	E4	–	–	–
EVReady	–	E4	–	E4	–	–	–
EVRESSConditioning	–	E4	–	E4	–	–	–
EVErrorCode	–	E4	–	E4	–	–	–
EVRESSS	–	E4	–	E4	–	–	–
DC							
EVTargetCurrent	–	E4	–	E4	–	–	–
EVMaximumVoltageLimit	–	E4	–	E4	–	–	–
EVMaximumCurrentLimit	–	E4	–	E4	–	–	–
EVMaximumPowerLimit	–	E4	–	E4	–	–	–
BulkChargingComplete	–	E4	–	E4	–	–	–
ChargingComplete	–	E4	–	E4	–	–	–
RemainingTimeToFullSoC	–	E4	–	E4	–	–	–
RemainingTimeToBulkSoC	–	E4	–	E4	–	–	–
EVTargetVoltage	–	E4	–	E4	–	–	–
CurrentDemandRes	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WPS
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–
Body	–	E4	–	E4	–	–	–
ResponseCode	–	E4	–	E4	–	–	–
DC_EVSEStatus	–	E4	–	E4	–	–	–
EVSEIsolationStatus	–	E4	–	E4	–	–	–
EVSEStatusCode	–	E4	–	E4	–	–	–
NotificationMaxDelay	–	E4	–	E4	–	–	–
EVSENotification	–	E4	–	E4	–	–	–
EVSEPresentVoltage	–	E4	–	E4	–	–	–
EVSEPresentCurrent	–	E4	–	E4	–	–	–
EVSECurrentLimitAchieved	–	E4	–	E4	–	–	–
EVSEVoltageLimitAchieved	–	E4	–	E4	–	–	–
EVSEPowerLimitAchieved	–	E4	–	E4	–	–	–
EVSEMaximumVoltageLimit	–	E4	–	E4	–	–	–
EVSEMaximumCurrentLimit	–	E4	–	E4	–	–	–
EVSEMaximumPowerLimit	–	E4	–	E4	–	–	–
EVSEID	–	–	–	E4	–	–	–
SAScheduleTupleID	–	–	–	E4	–	–	–
MeterInfo	–	–	–	E4	–	–	–
MeterID	–	–	–	E4	–	–	–

Продолжение таблицы К.1

	Зарядка переменным током с EIM	Зарядка постоянным током с EIM	Зарядка переменным током с PnC	Зарядка постоянным током с PnC	Опция: Обновление сертификата	Опция: Установка сертификата	Опция: WS
MeterReading	–	–	–	E4	–	–	–
BigMeterReading	–	–	–	E4	–	–	–
MeterStatus	–	–	–	E4	–	–	–
Tmeter	–	–	–	E4	–	–	–
ReceiptRequired	–	–	–	E4	–	–	–
WeldingDetectionReq	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature	–	–	–	–	–	–	–
Body	–	E4	–	E4	–	–	–
DC_EVStatus	–	E4	–	E4	–	–	–
EVReady	–	E4	–	E4	–	–	–
EVRESSConditioning	–	E4	–	E4	–	–	–
EVErrorCode	–	E4	–	E4	–	–	–
EVRESSSOC	–	E4	–	E4	–	–	–
WeldingDetectionRes	–	E4	–	E4	–	–	–
Header	–	E4	–	E4	–	–	–
SessionId	–	E4	–	E4	–	–	–

Окончание таблицы К.1

	Зарядная передающая точка ⇔ EVA	Зарядная приемная точка ⇔ EVA	Зарядная передающая точка ⇔ EVC	Зарядная приемная точка ⇔ EVC	Опция: Обязательно поддерживается	Опция: Устаревшее поддерживается	Опция: VAS
Notification	–	E4	–	E4	–	–	–
FaultCode	–	E4	–	E4	–	–	–
FaultMsg	–	E4	–	E4	–	–	–
Signature (см. отдельную таблицу)	–	–	–	–	–	–	–
Body	–	E4	–	E4	–	–	–
ResponseCode	–	E4	–	E4	–	–	–
DC_EVSEStatus	–	E4	–	E4	–	–	–
EVSEIsolationStatus	–	E4	–	E4	–	–	–
EVSEStatusCode	–	E4	–	E4	–	–	–
NotificationMaxDelay	–	E4	–	E4	–	–	–
EVSENotification	–	E4	–	E4	–	–	–
EVSEPresentVoltage	–	E4	–	E4	–	–	–

Приложение ДА
(справочное)

Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте

Таблица ДА.1

Обозначение ссылочного национального стандарта	Степень соответствия	Обозначение и наименование ссылочного международного стандарта
ГОСТ Р МЭК 61851-1—2013	IDT	IEC 61851-1:2010 «Система токопроводящей зарядки электромобилей. Часть 1. Общие требования»
ГОСТ Р МЭК 62196-1—2013	IDT	IEC 62196-1:2011 «Вилки, штепсельные розетки, соединители и вводы для электромобилей. Кондуктивная зарядка электромобилей. Часть 1. Общие требования»
ГОСТ Р МЭК 62196-2—2013	IDT	IEC 62196-2:2011 «Вилки, штепсельные розетки, соединители и вводы. Кондуктивная зарядка электрических транспортных средств. Часть 2. Требования к размерной совместимости и взаимозаменяемости для штыревых контактов и трубчатой контактной арматуры для переменного тока»

Примечание — В настоящей таблице использовано следующее условное обозначение степени соответствия стандартов:
- IDT — идентичные стандарты.

Приложение ДБ
(справочное)

Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта

Таблица ДБ.1

Структура настоящего стандарта		Структура международного стандарта ISO 15118:2—2014	
Приложения	A	Приложения	K
	B		F
	C		E
	Д		G
	E		J
	F		C
	G		H
	H		I
	I		B
	J		Д
	K		A
	ДА		—
	ДБ		—

Примечания

1 Сопоставление структуры стандартов приведено, начиная с приложений, так как в основной нормативной части настоящего стандарта структура не менялась.

2 Внесены дополнительные приложения ДА и ДБ в соответствии с требованиями, установленными в ГОСТ Р 1.7.

Библиография

[1] *ИСО 15118-3* — *Дорожные транспортные средства. Интерфейс связи транспортного средства и электросети. Часть 3. Требования к физическому и канальному уровню (Road vehicles — Vehicle to grid communication interface — Part 3: Physical and data link layer requirements)*

[2] ИСО 10731 — Информационные технологии. Взаимосвязь открытых систем. Базовая эталонная модель. Соглашения, касающиеся определения услуг в OSI (Information technology — Open Systems Interconnection — Basic Reference Model — Conventions for the definition of OSI services)

[3] Руководство Altova XMLSpy Manual [состояние на 2011-01-12], размещено на http://manual.altova.com/XMLSpy/spvprofessional/index.html?xseditinqviews schview contmodview.htm (Altova XMLSpy Manual [viewed 2011-01-12], Available from, gost.e-ecolog.ru)

[4] IETF RFC 5280 — Сертификат инфраструктуры открытых ключей Internet X.509 и профиль перечня аннулированных сертификатов (CRL) (май 2008 г.) [Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile (May 2008)]

[5] *IETF RFC 6960* — *X.509 Протокол статуса онлайнового сертификата инфраструктуры открытых ключей Интернета. OCSP (июнь 2013 г.) [X.509 Internet Public Key Infrastructure Online Certificate Status Protocol — OCSP (June 2013)]*

[6] Репозитарий идентификаторов объектов (ИО) [состояние на 2011-01-12], размещен на gost.e-ecolog.ru com/ (Object Identifier (OID) Repository [viewed 2011-01-12]. Available from <gost.e-ecolog.ru >)

[7] *NIST FIPS PUB 180-4* — *Стандарт защищенного хеширования (SHS) (март 2012 г.) [Secure Hash Standard (SHS) (March 2012)]*

[8] *МЭК 61851-22* — *Проводная зарядная система электрического транспортного средства. Часть 22. Станция для зарядки электрических транспортных средств переменным током (Electric vehicle conductive charging system — Part 22: AC electric vehicle charging station)*

[9] *МЭК 61851-23* — *Проводная зарядная система электрического транспортного средства. Часть 23. Станция для зарядки электрических транспортных средств постоянным током (редакция 1.0 2012) [Electric vehicle conductive charging system — Part 23: D.C. electric vehicle charging station (Ed 1.0 2012)]*

[10] *ИСО 17409* — *Транспорт дорожный на электрической тяге. Подсоединение к наружному источнику электропитания. Требования безопасности (Electrically propelled road vehicles — Connection to an external electric power supply — Safety requirements)*

[11] *IETF RFC 2460* — *Спецификация интернет-протокола, версия 6 (IPv6) (декабрь 1998 г.) [Internet Protocol, Version 6 (IPv6) Specification (December 1998)]*

[12] *IETF RFC 5095* — *Нежелательность маршрутных заголовков типа 0 в IPv6 (декабрь 2007 г.) [Deprecation of Type 0 Routing Headers in IPv6 (December 2007)]*

[13] IETF RFC 5871 — Указания IANA по назначению заголовков маршрутизации IPv6 (май 2010 г.) [IANA Allocation Guidelines for the IPv6 Routing Header (May 2010)]

[14] *IETF RFC 1981* — *Определение максимального размера передаваемого блока сообщения для сети интернет-протокола версии 6 (август 1996 г.) [Path MTU Discovery for IP version 6 (August 1996)]*

[15] *IETF RFC 5722* — *Обработка накладывающихся фрагментов IPv6 (декабрь 2009 г.) [Handling of Overlapping IPv6 Fragments (December 2009)]*

[16] IETF RFC 5220 — Формулировка задачи для выбора адреса по умолчанию в многопрефиксных средах: Операционные вопросы правил по умолчанию RFC 3484 (июль 2008 г.) [Problem Statement for Default Address Selection in Multi-Prefix Environments: Operational Issues of RFC 3484 Default Rules (July 2008)]

[17] *IETF RFC 3315* — *Протокол динамической конфигурации хостов для IPv6 (DHCPv6) (июль 2003 г.) [Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (July 2003)]*

[18] *IETF RFC 4861* — *Обнаружение соседних объектов для интернет-протокола версии 6 (IPv6) (сентябрь 2007 г.) [Neighbor Discovery for IP version 6 (IPv6) (September 2007)]*

[19] *IETF RFC 4429* — *Оптимистическое определение адресов-дубликатов (DAD) для IPv6 (апрель 2006 г.) [Optimistic Duplicate Address Detection (DAD) for IPv6 (April 2006)]*

[20] *IETF RFC 4443* — *Протокол управления сообщениями в сети Интернет (ICMP v6) для спецификации интернет-протокола версии 6 (IPv6) (март 2006 г.) [Internet Control Message Protocol (ICMP v6) for the Internet Protocol version 6 (IPv6) specification (March 2006)]*

[21]	IETF RFC 3122	Дополнения к IPv6 по обнаружению соседних объектов в части спецификации обратного обнаружения (июнь 2001 г.) [Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification (June 2001)]
[22]	IETF RFC 4291	Архитектура адресации интернет-протокола версии 6 (февраль 2006 г.) [IP Version 6 Addressing Architecture (February 2006)]
[23]	IETF RFC 4862	Автоматическая настройка адресов без сохранения состояния IPv6 (сентябрь 2007 г.) [IPv6 Stateless Address Autoconfiguration (September 2007)]
[24]	IETF RFC 6106	Варианты объявления маршрутизатора IPv6 для конфигурации DNS (ноябрь 2010 г.) [IPv6 Router Advertisement Options for DNS Configuration (November 2010)]
[25]	IETF RFC 3484	Выбор адресов по умолчанию для интернет-протокола версии 6 (IPv6) (февраль 2003 г.) [Default Address Selection for Internet Protocol version 6 (IPv6) (February 2003)]
[26]	IETF RFC 793	Протокол управления передачей. Программа Интернет DARPA. Спецификация протокола (сентябрь 1981 г.) [Transmission Control Protocol — DARPA Internet Program — Protocol Specification (September 1981)]
[27]	IETF RFC 5681	Контроль перегрузки TCP (сентябрь 2009 г.) [Congestion Control (September 2009)]
[28]	IETF RFC 6582	Изменение NewReno к алгоритму быстрого восстановления TCP (апрель 2012 г.) [The NewReno Modification to TCP's Fast Recovery Algorithm (April 2012)]
[29]	IETF RFC 6298	Вычисление таймера повторной передачи TCP (июнь 2011 г.) [Computing TCP's Retransmission Timer (June 2011)]
[30]	IETF RFC 1323	Расширения TCP для повышения производительности (май 1992 г.) [TCP Extensions for High Performance (May 1992)]
[31]	IETF RFC 2018	Опции селективных подтверждений TCP (октябрь 1996 г.) [TCP Selective Acknowledgment Options (October 1996)]
[32]	IETF RFC 5482	Опция тайм-аута пользователя TCP (март 2009 г.) [TCP User Timeout Option (March 2009)]
[33]	IETF RFC 1624	Вычисление контрольных сумм в Интернете посредством инкрементного обновления (май 1994 г.) [Computation of the Internet Checksum via Incremental Update (May 1994)]
[34]	IETF RFC 768	Протокол пользовательских дейтаграмм (август 1980 г.) [User Datagram Protocol (August 1980)]
[35]	IETF RFC 5246	Протокол безопасности транспортного уровня (TLS) версии 1.2 (август 2008 г.) [The Transport Layer Security (TLS) Protocol Version 1.2 (August 2008)]
[36]	IETF RFC 6066	Расширения протокола безопасности транспортного уровня (TLS): определения расширений (январь 2011 г.) [Transport Layer Security (TLS) Extensions: Extension Definitions (January 2011)]
[37]	IETF RFC 6961	Расширение протокола безопасности транспортного уровня (TLS) о запросе статуса нескольких сертификатов (июнь 2013 г.) [The Transport Layer Security (TLS) Multiple Certificate Status Request Extension (June 2013)]
[38]	IETF RFC 5289	Наборы шифров TLS на основе эллиптических кривых с SHA-256/384 и режимом счетчика Галуа (GCM) AES (август 2008 г.) [TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM) (August 2008)]
[39]	IETF RFC 6335	Процедуры службы назначения номеров Интернет (IANA) для управления реестром имен сервисов и номеров портов транспортного протокола (август 2011 г.) (Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transfer Protocol Port Number Registry (August 2011)
[40]	W3C EXI 1.0	Формат эффективного XML-обмена (EXI) 1.0. Рекомендация W3C (март 2011 г.) [(Efficient XML Interchange (EXI) Format 1.0, W3C Recommendation (March 2011)]
[41]	W3C XML Signature	Синтаксис и обработка подписи XML, версия 1.1. Рекомендация W3C (апрель 2013 г.) [Signature Syntax and Processing Version 1.1, W3C Recommendation (April 2013)]
[42]	W3C EXI Profile	Профиль W3C EXI, Формат эффективного XML-обмена (EXI) 1.0. Проект рекомендации W3C (июль 2013 г.) [W3C EXI Profile, Efficient XML Interchange (EXI) Profile, W3C Candidate Recommendation (July 2013)]
[43]	Специальная публикация NIST 800-38A	Рекомендация для режимов блочного шифрования. Методы и приемы (2001 г.) [Recommendation for Block Cipher Modes of Operation — Methods and Techniques (2001)]

[44] *Специальная публикация NIST 800-56A* — *Рекомендация по парным схемам создания ключей с помощью криптографии на базе дискретных логарифмов (переработанная) (март 2007 г.) (Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised) (March 2007))*

[45] *IETF RFC 5480* — *Криптография на основе эллиптических кривых с учетом информации об открытых ключах (март 2009 г.) [Elliptic Curve Cryptography Subject Public Key Information (March 2009)]*

[46] NIST SP 800-90 A — Рекомендация по генерированию случайных чисел с использованием детерминированных генераторов случайных двоичных последовательностей (январь 2012 г.) [Recommendation for Random Number Generation Using Deterministic Random Bit Generators (January 2012)]

[47] BSI AIS 20/AIS 31 — Предложение по: Классам функциональности для генераторов случайных чисел (2011 г.) [A proposal for: Functionality classes for random number generators (2011)]

[48] IETF RFC 1630 — Универсальные идентификаторы ресурсов во Всемирной сети (июнь 1994 г.)

[49] *IEC 62196-3* — *Вилки, штепсельные розетки, соединители и вводы для электромобилей. Кондуктивная зарядка электромобилей. Часть 3. Требования к совместимости размеров и взаимозаменяемости соединительных муфт со штырями и трубчатых муфт на постоянный и переменный/постоянный ток (Plugs, socket-outlets, vehicle connectors and vehicle inlets — Conductive charging of electric vehicles — Part 3: Dimensional compatibility and interchangeability requirements for d.c. and a.c./d.c. pin and contact-tube vehicle couplers)*

[50] *Реестр сервисов и портов IANA, реестр имен сервисов и номеров портов транспортного протокола [состояние на 2011-01-16], размещен на: http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml (IANA Service&PortRegistry, Service Name and Transport Protocol Port Number Registry [viewed 2011-01-16]. Available from:* gost.e-ecolog.ru *service-names-port-numbers.xml)*

УДК 629.3:006.354 ОКС 43.120

Ключевые слова: транспорт дорожный, интерфейс связи автомобиль — электрическая сеть, требования, протокол сетевого и прикладного уровней

БЗ 7—2018/108

Редактор *Н.Н. Кузьмина*
Технический редактор *И.Е. Черепкова*
Корректор *Е.Р. Ароян*
Компьютерная верстка *Ю.В. Половой*

Сдано в набор 03.10.2018. Подписано в печать 14.11.2018. Формат 60 × 84 1/8. Гарнитура Ариал.
Усл. печ. л. 39,53. Уч.-изд. л. 35,58.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ИД «Юриспруденция», 115419, Москва, ул. Орджоникидзе, 11.
www.jurisizdat.ru gost.e-ecolog.ru

Создано в единичном исполнении ФГУП «СТАНДАРТИНФОРМ»
для комплектования Федерального информационного фонда стандартов,
117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru gost.e-ecolog.ru

gost.e-ecolog.ru