IBM Research GmbH

Saeumerstrasse 4
CH - 8803 Rüschlikon
Switzerland

Subject: IBM FERN Smart Charging Interface between EV FO and Retailer (Draft)

Version: 1.1
Date: July 30, 2012

Authors: Aditya Mishra, Dieter Gantenbein, Bernhard Jansen

1. Introduction

This document proposes a communication interface between an EV (Electric Vehicle) FO (Fleet Operator) and an electricity Retailer to enable smart charging. The interface is aligned with the guidelines given in ISO IEC 15118 ([1] [2]) for communication between an EVSE (Electric Vehicle Supply Equipment) and a SA (Secondary Actor). Based on the information received from the FO Retailer responds with an optimized charging schedule for the EV. Data structures, satisfying the stated requirements, for sending information from FO to the retailer and from Retailer to the FO are outlined. Focus of this document is to outline data structures for enabling smart charging of EVs, it does not address billing and metering requirements.

Our approach would further facilitate electric grid regulation on top of power link modem. Being a B2B communication protocol between a FO and an electricity Retailer it is an orthogonal and complementary approach to the proposals made in ISO IEC 15118.

OCPP (Open Charge Point Protocol) proposes an alternative way of enabling smart charging for the EVs [7]. OCPP assumes that the EVSEs are controlled/managed by a central system which schedules charging of the EVs. In terms of ISO IEC 15118, OCPP implements communication between EVSE and the SA (Secondary Actor). There can be multiple ways of extending/implementing the recommendations of 15118 for EV smart charging. In contrast to OCPP, our approach enables smart charging by enabling B2B communication between a FO and the Retailer, independent of any EVSE management system. We propose an architecture complementary to ISO IEC 15118 for enabling EV trip prediction ([8]) and flexible charging optimization for EVs [9].

2. Simplified Interaction Diagrams

2.1 Context Diagram

Figure 1 shows a simplified context diagram for EV smart charging system. It outlines the main actors in the system and communication protocols used for message exchange between them. Following is a short summary about each of the actors shown in figure 1.

- DCH (Demand Clearing House): An entity for grid negotiation that provides information on the load of the grid. It mediates between two clearing partners – a SECC (Supply Equipment Communication Controller) and the part of the power grid connected to this SECC [2].
- EV Owner: An individual or an entity which owns the EV.
- EV: Vehicle propelled by an electric motor drawing current from a rechargeable storage battery or from other portable energy storage devices, which is manufactured primarily for use on public streets, roads or highways [2].
- EVSE (Electric Vehicle Supply Equipment): Conductors, including the phase(s), neutral and protective earth conductors, the EV couplers, attached plugs, and all other accessories, devices, power outlets or apparatuses installed specifically for the purpose of delivering energy from the premises wiring to the EV and allowing communication between them as necessary [2].
- FO: An individual or an entity operating several EVs.

- OEM (Original Equipment Manufacturer): It manufactures products or components that are purchased by a company and retailed under the purchasing company's brand name. OEM refers to the company that originally manufactured the component [2].
- Retailer: The body which provides electricity.

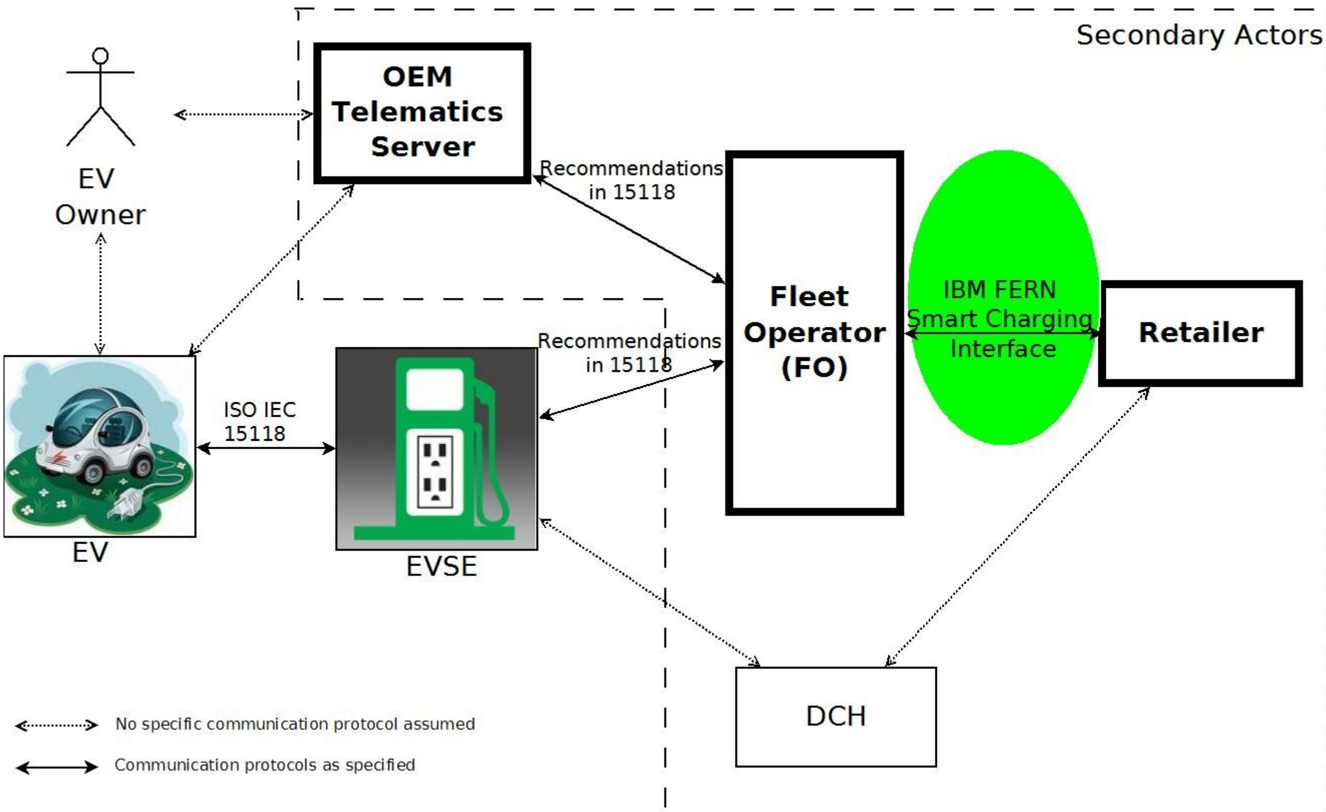

Figure 1: Context Diagram for EV smart charging

2.2 Sequence Diagram

Figure 2 shows a simplified sequence diagram depicting message exchange between the participating actors for EV smart charging. Objects (or actors) are the same as defined in section 2.1. Note that the scenario shown in figure 2 is only an illustrative example and in practice the 'evse:EVSE' can potentially be a “virtual” evse, i.e., functioning as an EVSE but could be located either in the socket or in the EV. However, this does not affect the proposed data structure for 'ChargeRequestMessage' given in section 4.1.2. Most of the messages shown in the figure are intuitive and self-explanatory. In the following we give explanation of objects and functions not covered before.

- GridSchedule: Maximum charging current vs. time information, which is possible at the current outlet based on the local grid situation [2]. Local grid demand and supply situation (both actual and forecast) are some of the inputs to calculate grid schedule.

- LevelSelector: Function to select the lowest value among the Sales tariff table, Grid schedule and local physical limits [2].
- ChargingPreference: An enumeration which allows the EV to specify its preference about the charging mode/method for the given charging session. Charging modes can be 'Immediate Charging', 'SLA Charging', 'Green Charging', etc..
- ProfileLookup: Function call executed at the FO to check the charging modes that a given EV is authorized to request according to its contract.

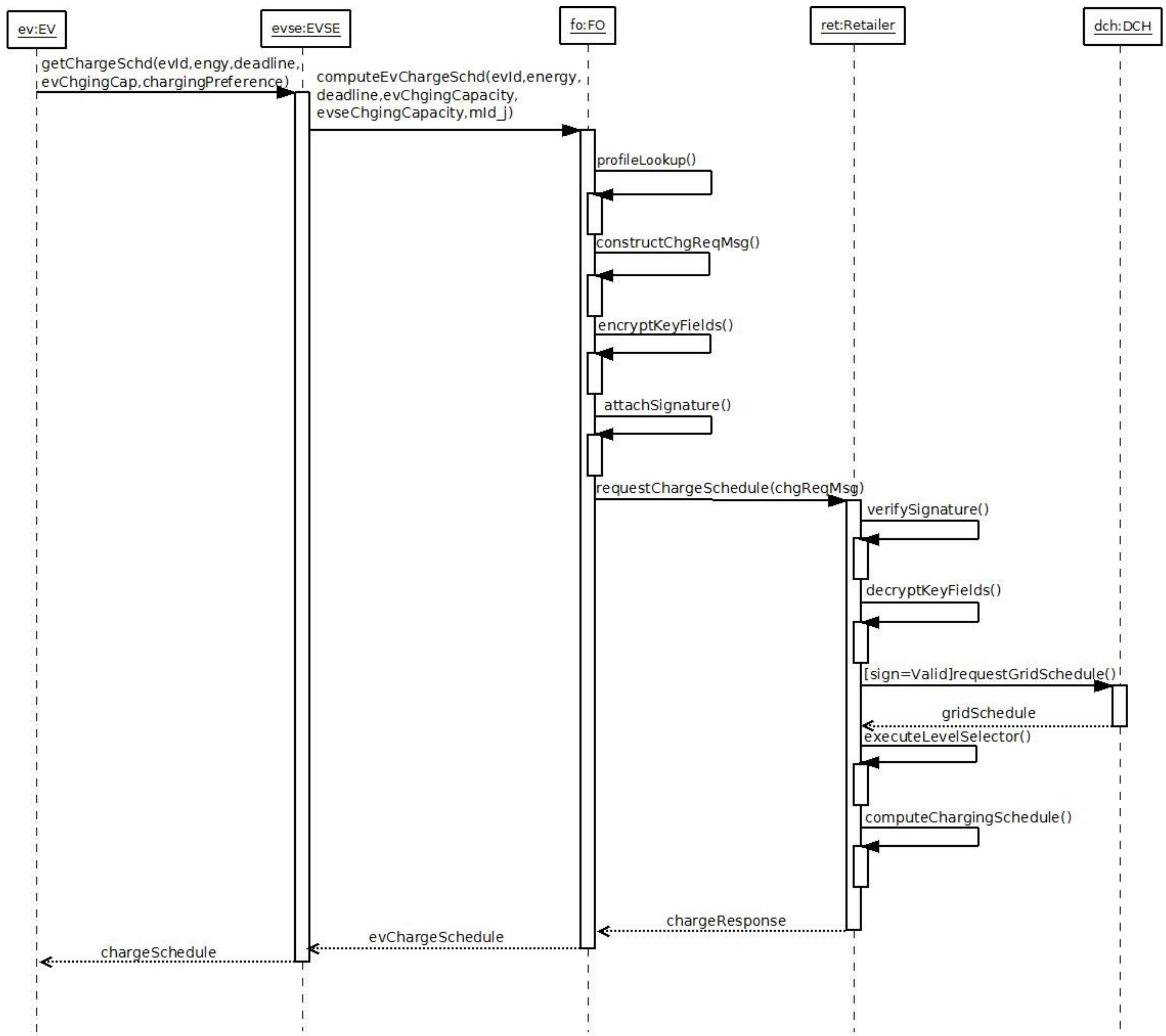

Figure 2: Simplified sequence diagram for EV smart charging request

3. Requirements

3.1 Data Requirements

In order to enable an effective exchange of information between FO and a retailer following attributes need to be exchanged in the indicated direction.

3.1.1 FO to Retailer

A charging request consists of the following fields:-

- Timestamp indicating the time when request originated.
- A meter ID, possibly containing a subindex to address the fact that multiple charging sockets might be supported on a single meter.
- Reference ID to uniquely identify the requesting EV.
- Maximum charging power that is supported between the EV and EVSE at the current charging spot at this point in time.
- Unit of power, e.g., W, kW, MW, etc.
- Minimum time for which the EV should be powered before turning OFF.
- Minimum time for which the EV should remain OFF before powering ON again.
- A set of charging request elements consisting of any number of charging request elements comprising:-
  - Start time indicating the expected time when charging is expected to start.
  - End time indicating the deadline by when the EV should get the minimum amount of charge requested.
  - Minimum energy needed, this might be the energy required for the next trip.
  - Maximum energy, this indicates the total left over capacity of the EV battery.
  - Unit of energy, e.g., J, Wh, kWh, etc.

3.1.2 Retailer to FO

- Reference ID to uniquely identify the requesting EV.
- The energy unit for EV charging.
- The power unit for EV charging.
- Set of charging schedules satisfying the constraints stated in the charging request. Details of the charging schedule are given in section 3.1.3.

3.2 Security Requirements

Following security requirements need to be assured for securing the communication between FO and retailer.

- Authenticity – This ensures the trustfulness of the origin of the information.
- Integrity – The information has not been altered on its way to the destination.
- Non-repudiation – The source cannot deny having sent the information.

- Non-replay able – The information cannot be recorded and replayed at a different point in time by an attacker.
- Confidentiality - Preventing the disclosure of information to unauthorized individuals or systems.

4. Data Structures

4.1 Message format and encoding

To allow an easy adoption of transport protocols the application level protocol should be independent of transport layer protocol. All the stated data and security requirements are covered by the application layer message format proposed in this section.

4.1.1 Message encoding

The data described in section 2.1 is represented as a XML structure, which is described as an XML schema.

The data structures are generalized and allow for information exchange between a diverse set of FOs and retailers. It allows for sending as many charging request elements, spread over time, as required. This gives EV/FO the flexibility to have greater control over charging. All the times are given in UTC. Reference Id of the EV can either be the original vehicle ID or a reference to it (mapped at the FO). Inclusion of maximum energy enables the retailer to provide any surplus energy to EV.

4.1.2 FO to Retailer message format

A FO (Fleet Operator) sends ChargeRequestMessage to the Retailer, requesting charging schedules for the EV. A ChargeRequestMessage contains a list of contiguous (in time) ChargeRequestElements. Each ChargeRequestElement specifies the specific energy and power needs of the EV in a given time window. UML representation of the message is shown in figure 3. ChargeRequestMessage contains the following fields:-

- timestamp: This indicates the time at which message originated.
- ttl: Time To Live, this is the duration (since timestamp) for which the message is valid. It helps countering replay attacks as mentioned in section 4.2.
- $meterId_j$: This is the electric meter ID. It serves as an EV location indicator.
- reference: This unique reference is used to identify the EV in a charging session. Vehicle ID can be used as a reference. However, if there are privacy concerns then FO can map the vehicle ID to a temporary reference valid for a single charging session.
- maxPower: This is the maximum power that can be supported between the EV and EVSE at current outlet.
- pow: The unit for measuring power, e.g., Watt, kWatt, etc.
- minOnTime: This is the minimum time for which the EV should be powered before turning OFF, to prevent damages to battery life due to frequent ON-OFF events.

- minOffTime: Minimum time for which the EV should remain OFF before powering ON again.
- tim: Unit of time for minOnTime and minOffTime.
- A list of ChargeRequestElement. Each ChargeRequestElement has the following fields :-
    - startTime: This is the starting time for this ChargeRequestElement time window.
    - endTime: This is the end time for this ChargeRequestElement time window.
    - minEnergy: This is the minimum amount of energy which the EV needs in the interval between startTime and endTime. If the value in this field is positive then the energy is requested by the EV for charging. However, if the value is negative, the EV has extra charge, equal to the magnitude of the value, which it can feed into the grid. Note that the sum of the 'minEnergy' values across all ChargeRequestElement items should be less than or equal to the total remaining capacity of the EV at the time when charging request is made.
    - maxEnergy: This value indicates the maximum energy which can be fed into the EV during the interval between startTime and endTime. Again, if the value is positive it means energy is requested for charging the EV, and a negative value implies energy can be fed from the EV into the grid.
    - engy: Unit of energy, e.g., Joule, Wh, kWh, etc.

Contents of the message are signed by the FO and signature is attached to the message. This ensures that the Retailer can verify the authenticity and integrity of the message. Moreover, in order to protect the privacy of the EV the meterID and ev_reference are encrypted (using Retailer's public key).

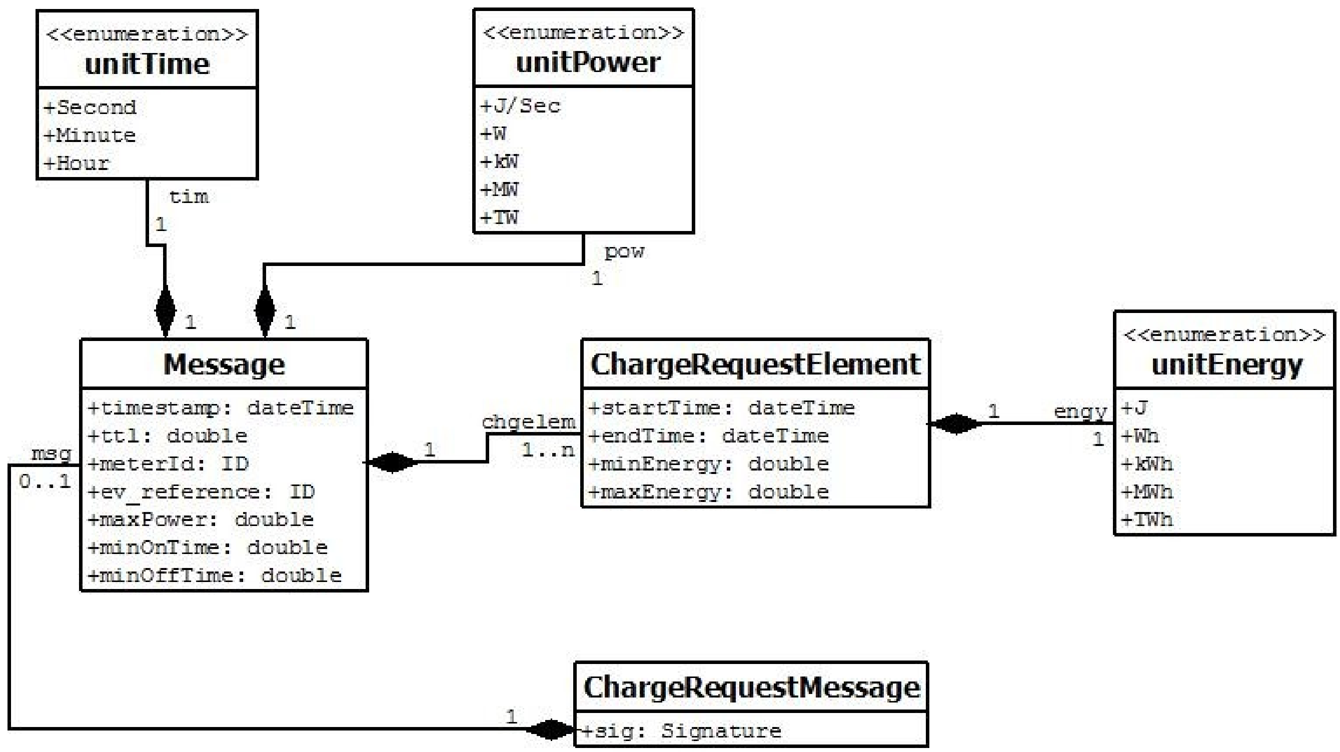

Figure 3: UML representation of message from FO to the Retailer

4.1.3 Retailer to FO message format

Format of the response message containing proposed charging schedules for the EV from the Retailer to the FO has been adopted from the 'SAScheduleListType' defined in ISO IEC 15118 [1]. ISO IEC 15118 is mainly focused on communication between an EV and an EVSE. Adopting the response message format between Retailer and FO from ISO IEC 15118 will facilitate interoperability. Figure 4 shows UML diagram representation of the message format.

Following is a short summary of different fields and objects in the response message.

1. ChargeResponse – This is the response object sent by Retailer and contains the proposed charging schedules for the EV. It contains,
   - timestamp: The time at which response originates.
   - ev_reference: This is the ID used for identifying the requesting EV.
   - sig: Signature attached to the message for verifying message authenticity and integrity.
   - engy: Energy unit for all the energy fields in the message.
   - pow: Power unit for all the power fields in the message.
   - ScheduleList: A list of proposed charging schedules from the retailer. Its a collection of ScheduleTuple elements. The first ScheduleTuple element in the list is defined as the default schedule. An EVCC may have a mechanism of comparing different ScheduleTuple elements and could choose an optimal charge schedule according to a cost. However, if no

such mechanism exists in the EVCC then the default schedule can be used.

2. ScheduleTuple – Each ScheduleTuple represents a proposed schedule. It consists of the following
   - ScheduleTupleID: This is a unique identifier within a charging session for identifying the ScheduleTuple.
   - PMaxSchedule: Description below.
   - SalesTariff: Description below.
3. PMaxSchedule – This element contains all relevant details for one PMaxSchedule from the retailer. It consists of:-
   - PMaxScheduleID: Unique identifier across a charging session.
   - PMaxScheduleEntry: This is a list of PMaxScheduleEntry elements. Each PmaxScheduleEntry is composed of,
     - RelativeTimeInterval: Defining the start time and duration of the period for this entry. More details below.
     - PMax: This is the maximum amount of power to be drawn from EVSE outlet to which the EV is connected during the specified time interval.
4. SalesTariff – This element contains all relevant details for one SalesTariff from the retailer. The sales tariff table enables EVCC to optimize its charge schedule based on some cost. A SalesTariff consists of the following:-
   - SalesTariffID: A unique ID used for identifying a given sales tariff.
   - SalesTariffDescription: This string contains a human readable description of the sales tariff.
   - NumEPriceLevels: This defines the maximum number of distinct price levels across all the tariffs provided. For example, on-peak, mid-peak, off-peak means three price levels. It is the overall number of distinct EPriceLevels across all SalesTariff elements in the list of ScheduleTuple. This allows for cost optimization and comparison between different tariffs.
   - SalesTariffEntry: Description below.
5. SalesTariffEntry – This element describes all relevant details for one time interval of the SalesTariff. It contains the following fields:-
   - RelativeTimeInterval: Defining the start time and duration of the period for which this entry is valid. More details below.
   - EPriceLevel: Defines the price level of this SalesTariffEntry (with respect to NumEPriceLevels). Small values means cheaper price entries and large values imply more expensive tariff.
   - ConsumptionCost: This is a list of ConsumptionCostType elements.
6. ConsumptionCost – It defines additional means for specifying relative price information and/or alternative costs. It contains:-
   - startValue: The lowest level of consumption that defines the starting point of this consumption block. The interval ends at the start of the next interval. The first element in the list of ConsumptionCost elements should start with a startValue set to 0.
   - Cost: Description below.
7. Cost – Describes all relevant cost details for this consumption block in this TariffEntry. It contains:-
   - amount: The estimated or actual cost, per unit of measurement (typically

kWh).
    - amountMultiplier: This is an optional decimal shift operator for amount. It is defined as the exponent to base 10.
    - costKind: Description below.
8. CostKind – This defines the kind of cost referred to in the message element amount. This is an enumeration and defines the following three types of costs:-
    - relativePercentagePrice: Relative price, as percentage relative to a common base between all tariffs. For this kind of cost the amount element is defined as percentage relative to the highest cost of the same kind.
    - RenewableGenerationPercentage: Renewable generation as a percentage of overall generation.
    - CarbonDioxideEmission: CO_2 emissions in grams per unit of measure.
9. RelativeTimeInterval – It has the following two fields:-
    - duration: Duration of the interval in seconds
    - start: Start time of the interval, defined in seconds from NOW. In Unix Time Stamp format.

Retailer signs the message and the signature is attached to the response. FO can verify the integrity and authenticity of response message using this signature. Further, in order to keep the identity of EV secret the ev_reference is encrypted using public key of the FO.

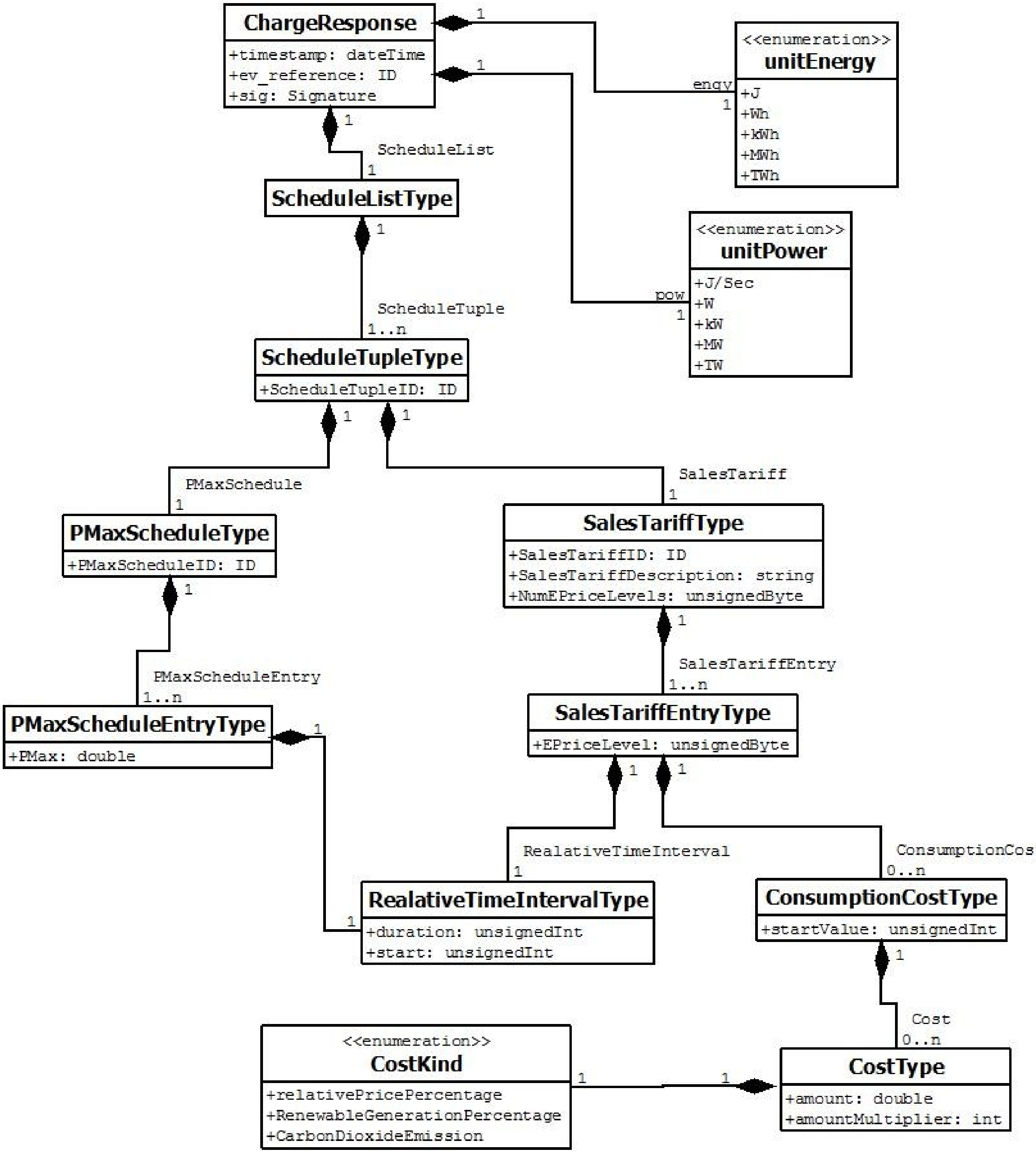

Figure 4: UML representation of message from Retailer to the FO

4.2 Security

The security requirements outlined in section 2.2 are met by using signature and encryption of the sensitive/private message fields using PKI infrastructure. For message signature schema the standardized W3C XML signature, encryption syntax and processing is used. The signature and cipher text provide the following security primitives.

- Authenticity – This is provided by the signature, as the signature contains the certificate of the signee which is signed by a trusted third party.
- Confidentiality – Key components of the message are encrypted using receivers public key, hence, it can be decrypted only by the receiver using its private key.
- Integrity – The signed hash of decrypted message can be recomputed and compared against the signed hash received with the message. Hence, it can be verified whether integrity of message is intact or not.
- Non-replay able – No message can be replayed as a valid message by an attacker. It would either be discarded as an outdated message by the receiver looking at the message timestamp and ttl fields or if the timestamp is modified the message would fail the integrity check.
- Non-repudiation – It is provided by the signature of signee (sender). Certificate with signee's identity is attached to the signed message.

5. References

[1] DRAFT INTERNATIONAL STANDARD ISO/IEC DIS 15118-2: Road vehicles — Vehicle to grid communication interface — Part 2: Network and application protocol requirements. 2012.

[2] DRAFT INTERNATIONAL STANDARD ISO/IEC DIS 15118-1: Road vehicles — Vehicle to grid communication interface — Part 1: General information and use-case definition. 2012.

[3] IBM and EKZ Make Electric Vehicle Charging More Convenient with New Smartphone Application. http://www-03.ibm.com/press/us/en/pressrelease/35627.wss, October 2011.

[4] SmartGrid. http://www.zurich.ibm.com/smartgrid/.

[5] The EDISON Project: Making electric vehicles an essential part of a smart grid. http://www-01.ibm.com/software/success/cssdb.nsf/CS/SSAO-8CZJFW?OpenDocument&Site=corp&cty=en_us, December 2011.

[6] EDISON WP3 Deliverable D3.1 - Distributed Integration Technology Development. http://www.edison-net.dk/Dissemination/Reports/D3.1.aspx, April 2011.

[7] OCPP v1.5 A functional description. http://www.ocpp.nl/images/files/ocpp%201%205%20-%20a%20functional%20description%20v2%200.pdf, 2012.

[8] Olle Sundstroem, Olivier Corradi, Carl Binding. Toward Electric Vehicle Trip Prediction for a Charging Service Provider. First IEEE International Electric Vehicle Conference - Greenville SC, March 4-8, 2012.

[9] Olle Sundstroem, Carl Binding. Flexible Charging Optimization for EVs considering Distribution Grid Constraints. IEEE Transactions on Smart Grid, Volume.99, p.1-12, 2012.