Open Charge Metering Format

ENTWURF

Revision: 1.0.1

Mitwirkende

Rolle	Verantwortlicher	Firma
Verfasser	Andreas Mull, ABL (AM), Daniel Müller (DM)	ABL
Mitwirkung	Gerhard Weidinger, Florian Riegler	KEBA
Mitwirkung	Martin Klässner, Roland Angerer	has.to.be
Mitwirkung	Andreas Weber	Allego
Mitwirkung	Michael Staubermann	Webolution

Revisionsübersicht

Änderungen gegenüber der vorherigen Version sind Änderungsnachverfolgung gekennzeichnet.

Revision	Inhalt	Datum
1.0.1	Definition der Bedeutung der Versionsnummern. Genauere Definition des Aufbaus den Formats. Löschen Binärformat und Übertragungsformat (Da eine Transformation nicht ohne Neuberechnung der Signatur möglich ist).	
1.0	Finaler Stand zur Integration in Transparenzsoftware. Feedback aus SAFE AG-Technik eingearbeitet: Signatur-Algorithmus doch nicht in Data-to-be-Signed-Bereich, damit theoretisch mehrere unterschiedliche Signaturen unabhängig voneinander angehängt werden können. Reading-Type optional. Zusätzlicher Zustand beim Status der Zeit: Relative Zeitabrechnung basierend auf Info-Uhr. OBIS-Codes erweitert auf anwendbare Möglichkeiten entsprechend Kommentaren von Herrn Weber.	21.02.2019/AM
0.5	Weiteres Feedback aus DKE/GAK 461.0.21 AP5 von Januar 2019 eingearbeitet: Klarstellungen zur Paginierung, Bezug von Seriennummern, Ladepunkt und Public-Key. Optionale Nutzdatensektion zur Zuordnung des Ladepunkts. Fehlerindex ersetzt durch fehlerhafte Größen; Fehler während Ladung als Ablesungsgrund. Feedback von Herrn Weber zu Version 0.3 eingearbeitet: Einspeißung, Stromarten. Byte-Offset bei den Binärbeschreibungen entfernt, da sich diese mit den StrEnum-Typen widersprachen. Feedback von Herrn Staubermann zur Elliptic-Curve-Kryptografie: Synonyme Namen der Kurven, Korrektur der Blocklängen bei den Public-Key-Typen.	11.02.2019/AM
0.4	Ergebnisse und Klarstellungen nach Feedback aus Telefonkonferenzen mit DKE/GAK 461.0.21 AP5: Dokument- und Format-Versionen korrigiert; Binärformat klargestellt: Network-Byte-Order; Zählung der Paginierung klargestellt, Umbruch definiert; Identifikation der Signatureinrichtung (Gateway) über Zähleridentifikation gestellt, in allgemeine Angaben integriert, Vendor zu Gateway umbenannt, Seriennummer entweder für Gateway oder Zähler obligatorisch; Benutzerzuordnung: Status aufgeteilt in generell ja/nein und Zuordungs-Level, Level in Gruppen eingeteilt; alle weiteren Angaben zum Status außer ja/nein sind nun optional. Verwendung Ereignisszähler klargestellt, Umbruch definiert; Hinweis auf Genauigkeit bei Messwert; Signatur-Algorithmus in den signierten Bereich verschoben; Signaturalgorithmen um zukünftige Kandidaten erweitert, Schlüsseltypen analog; Defaults mit OCMF-Version versehen. Autorisierungsmerkmale entsprechend OCPP 2.0 erweitert; Erweiterungspunkte für OCMF in Nutzdaten- und Signatursektion festgelegt.	11.01.2019/AM

0.3	Zustand des Zählers hinzugefügt für falsch ausgelesene Werte	11.12.2018/DM
0.2	Binärformat entsprechend der Umstellungen mit 0.1 angepasst, Beschreibung des Signatur-Algorithmus konkretisiert, StrEnum-Typ als flexiblen Erweiterungspunkt ins Binärformat eingeführt.	16.08.2018/AM
0.1	Dieses Dokument basiert auf ABL-Datenformat Version 1.14 und löst dieses ab. Initiale Version nach Diskussion des ABL-Datenformats (v1.13) mit KEBA und has.to.be: Entkopplung des binären Übertragungsformats aus dem eichrechtlich relevanten Teil, JSON als primäres Übertragungsformat. Header und Gesamtstruktur definiert. Felder abgeglichen mit dem Entwurfsstand „Anforderungen an die Datenstruktur für Messwerttupel in der Elektromobilität“ vom 26.04.2018. JSON-Format strukturell vereinfacht, Keys verkürzt. Möglichkeit mehrerer Ablesungen in einem Datensatz geschaffen. Firmware-Version des Zählers optional hinzugefügt. Fehler als Abbruch-Gründe für Ladevorgänge. Hash-Typ für Signatur definiert.	09.08.2018/AM

1 Allgemeines

1.1 Ziel

Ziel dieses Dokumentes ist es, ein unabhängiges und allgemein verwendbares Datenformat zur Erfassung eichrechtlich relevanter Zählerablesungen von Ladestationen zu beschreiben. Ferner kann dieses Konzept als Diskussionsgrundlage für eine Verwertung im DKE dienen. Außerdem soll es zur Implementierung der Auswertung und Signaturprüfung des Formats durch die im Rahmen der SAFE-Initiative zu schaffende Transparenz-Software dienen.

Zum Zeitpunkt der Erstellung dieses Dokuments ist kein einheitliches Datenformat zur Übermittlung von signierten Zählerständen auf Normungsebene verabschiedet. Eine entsprechende Anwendungsrichtlinie ist in Entstehung befindlich. Das vorgelegte Datenformat entstammt einem internen Entwurf von ABL und wurde mit has.to.be und KEBA diskutiert.

Ein einheitliches Format bringt den Vorteil der Interoperabilität und ermöglicht eine einheitliche Verifizierungs-Software beim Betreiber und dem Endbenutzer.

Das hier dargestellte Konzept orientiert sich an dem *Ende-zu-Ende Sicherungskonzept für eine eichrechtlich günstige Lösung („GL“) für die Übertragung von Messdaten auf Fernanzeigen* von Dr. Martin Kahmann [MEMO-GL]. Weiter baut es auf das ABL-interne Konzept zur Eichrechtskonformität auf.

Dieses Dokument ist wie folgt gegliedert: Zunächst wird eine Einordnung und Darstellung der Übermittlung des Datenformats gegeben, in den weiteren Sektionen wird das Datenformat erläutert.

1.2 Lizenz

TODOs:

- Lizenz definieren
- MIT: sehr frei, Verwertung in Werk enthaltener Patente nicht geregelt
- Apache: auch sehr frei, soll jedoch Regelung zu Patenten enthalten -> Klären!

1.3 Referenzen

OCPP 1.5: Open Charge Alliance: Open Charge Point Protocol – Interface description between Charge Point and Central System, Version 1.5, Stand: 8. Juni 2012

MEMO-GL: Messsysteme für Ladeeinrichtungen: Ende-zu-Ende-Sicherungskonzept für eine eichrechtlich günstige Lösung („GL“) für die Übertragung von Messdaten auf Fernanzeigen, Dr. Martin Kahmann, Braunschweig

OBIS: IEC 62056-6-1 und -6-2.

JSON: Einführung in JSON, https://www.json.org/json-de.html (Abruf: 2018-04-04)

1.4 Verwendete Abkürzungen

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CDR	Charge Data Record
CPO	Charge Point Operator, Betreiber der Ladestation
DKE	Deutsche Kommission Elektrotechnik Elektronik Informationstechnik
ECDSA	Elliptic Curve Digital Signature Algorithm
EMP	Electric Mobility Provider, Fahrstromanbieter
ISO	International Standards Organisation
MID	Measuring Instruments Directive, europäische Richtlinie 2004/22/EG
OBIS	OBject Identification System
OCPP	Open Charge Point Protocol
PIN	Personal Identification Number
PTB	Physikalisch-Technische Bundesanstalt
RFC	Request For Comments
RFID	Radio Frequency Identification
RTU	Remote Terminal Unit
SHA	Secure Hash Algorithm
TLS	Transport Layer Security
UID	Unique IDentification

UTC Universal Time Coordinated

1.5 Versionierung des Dokuments

Die Versionsnummer dieses Dokuments ist folgendermaßen:

Major Version:
Änderungen an dieser Stelle bedeuten, dass sich das Format grundlegend geändert hat. Es wurden Änderungen am Format vorgenommen die nicht mehr mit den Vorgängerversionen kompatibel sind.

Minor:
Das Format wurde ergänzt neue Felder bzw. die Werte eines Feldes wurden hinzugefügt (
Änderungen am Format. Noch abwärtskompatibel).

Revision:
Änderung an der Dokumentation. Keine Änderung am Format. Durch eine Erhöhung der Revision wird das Format an sich nicht geändert. Die Dokumentation wird angepasst bzw. verbessert.

Beispiel: 1.0.2
1 = Major
0 = Minor
2 = Revision

2 Übermittlung signierter Zählerstände

2.1 Einbettung in OCPP

Bereits OCPP Version 1.5 erlaubt im Rahmen der MeterValue.req und StopTransaction.req Nachrichten die simultane Übermittlung einer beliebigen Anzahl von MeterValue-Objekten. In jedem solchen Objekt kann wiederum ein Zeitstempel zusammen mit einem oder mehreren Ablesungswerten notiert sein. Jeder Wert kann optional durch Attribute begleitet werden, eines davon vom Enum-Typ ValueFormat, der zwei mögliche Werte besitzt:

- Raw: Data is to be interpreted as integer/decimal numeric data.
- SignedData: Data is represented as a signed binary data block, encoded as hex data.

Quelle: [OCPP 1.5]

Damit ist es möglich, die von OCPP vorgesehene Funktionalität weiterhin zu nutzen und mit der Möglichkeit der Übermittlung von signierten Zählerwerten zu kombinieren.

Das hier vorgestellte Datenformat ist folglich gedacht zur parallelen Übermittlung eines solchen SignedData-„Wertes“ neben den bisherigen Raw-Werten.

Während die Ablesung der bisherigen Werte evtl. mit einer nicht eichrechtskonformen (OCPP-basierten) Zeit geschieht, basieren die signierten Zählerwerte jedoch auf einer solchen. Sogar die zwei resultierenden Zeiten können getrennt in zwei MeterValue-Objekten korrekt dargestellt werden.

Dieser Mechanismus wird genutzt im Rahmen von:

- Beginn einer Transaktion (StartTransaction)
- Ablesungen während einer Transaktion (MeterValues)
- Ende der Transaktion (StopTransaction)
- Charge Data Record über die gesamte Transaktion mit Werten zu Beginn und Ende (optional)
- Ablesungen für sogenanntes „Fiscal-Metering“ (MeterValuesAlignedData): Dabei wird kein Bezug zu laufenden Transaktionen erfasst, sondern nur die absoluten Zählerwerte aller Ladepunkte zu vorher bestimmten Zeitpunkten.

Anzumerken ist, dass in der Regel eichrechtlich relevant folgende Ablesungen sind:

- Beginn- und Endwert in Bezug auf den Ladevorgang
- Tarifwechsel (gängige Praxis nur zur Viertelstunde der Uhr)

Damit ist folgende OCPP-Konfiguration typischerweise sinnvoll:

- ClockAlignedDataInterval=900 (15min)
- MeterValuesAlignedData=Active.Energy.Register.Import
- Eine Controller-Software sorgt für die Erzeugung zusätzlicher MeterValues mit signierten Werten zu Beginn und Ende des Ladevorgangs über den simplen Integer-Wert hinaus, der Bestandteil der StartTransaction/StopTransaction-Nachrichten sein kann.

2.2 Signierungs- und Verifikations-Prozess

Um einen Datensatz zu prüfen, müssen zunächst die Daten, welche signiert wurden,wieder in ihrer Ursprungsform hergestellt werden. Auf dieser Ursprungsform erfolgt eine Hash-Berechnung. Das Resultat wird über die digitale Signatur per ECDSA geprüft.

Zur Herstellung der Ursprungsform werden Header und Signatur vom Datensatz abgetrennt. Der Public-Key kommt ggf. über einen getrennten Übertragungsweg zur Prüfung. Damit sind im vereinheitlichten Datensatz, die Signatur und der Public-Key voneinander getrennt dargestellt. Der Zusammenhang zwischen mehreren einzelnen Datensätzen wird durch die durchgehende Paginierung gewährleistet. Diese ist neben der Signatur von einer Prüfkomponente zu verifizieren. Der erste Datensatz muss als Beginn eines Ladevorgangs, der letzte als Ende des Ladevorgangs gekennzeichnet sein. Dazwischen dürfen keine Datensätze entnommen oder hinzugefügt worden sein. Ebenso müssen zwischenzeitliche Fehlerzustände (Fehlerzähler) und das Auffinden von nicht verwendbaren Größen zu einem Fehler bei der Prüfung führen. Außerdem müssen alle Datensätze von derselben Quelle (Seriennummer) stammen.

3 Datenformat

Das Datenformat soll möglichst einfach aufgebaut und vom versierten Benutzer nachvollziehbar sein. Aus Herstellersicht ist ein modulares Datenformat wünschenswert, damit zukünftige Erweiterungen umgesetzt werden können. Ein weiteres Anliegen ist, den Prüfprozess möglichst generisch, d.h. auch für andere Quellformate anwendbar, zu gestalten. Aus diesem Grund wurde eine Header Sektion in das Format aufgenommen.

Das Format bestehet aus mehreren Sektionen:

1. **Header** zur eindeutigen Kennzeichnung des Formats
2. Abschnitt mit den eigentlichen **Nutzdaten**
3. **Signatur** über die Nutzdaten
4. Optional: Public-Key

Die Inhalte des Datensatzes dienen als Eingabeformat in die Transparenz-Software. Die Transparenz-Software ist online für jedermann verfügbar und kann zur Validierung des Datensatzes genutzt werden.

In den folgenden Kapiteln wird das Datenformat für die Darstellung und Übertragung der signierten Zählerwerte definiert.

4 JSON-basiertes OCMF Format

Für die einzelnen Sektionen des Formates wurde die JSON-Notation zur besseren Lesbarkeit und einfacheren Nachvollziehbarkeit gewählt. Um Datenvolumen einzusparen, wurden folgenden Maßnahmen getroffen:

- Bezeichner sind kurz gehalten (da einige ohnehin erklärungsbedürftig sind, geht hier nicht viel Klarheit verloren)
- An einigen Stellen wurde bewusst auf Kapselung verzichtet. Anstelle sind die Bezeichner kanonisiert.

Die Sektionen innerhalb des Datenformates werden mit dem Sonderzeichen „|“ getrennt. Innerhalb der Sektionen entspricht das Format der JSON-Notation. Somit ist die Nutzung des Sonderzeichens „|“ innerhalb der Sektionen nicht gestattet.

Dass das Format entsprechend der JSON-Notation mit Leerzeichen und Zeilenvorschüben an beliebiger Stelle, genannt White-Space, zur Darstellung angereichert werden kann erleichtert es einem Nutzer zudem die Lesbarkeit.

Zwischen Signierung und Validierung darf die Nutzdatensektion nicht manipuliert werden (entfernen und Hinzufügen von White-Spaces), da ansonsten eine positive Validierung nicht möglich ist.

Im Folgenden werden die verschiedenen Abschnitte des Formats erläutert. Weiterhin wird ein Beispiel gegeben.

4.1 Sektionen

Am Anfang des Formats steht ein Header zur eindeutigen Kennzeichnung des Formats.

Durch Pipe-Zeichen getrennt folgen Nutzdatensektion und Signatursektion.

Damit ergibt sich für die Sektionen folgender Aufbau:

OCMF|<Nutzdatensektion>|<Signatursektion>

Dabei sind die Worte Nutzdatensektion und Signatursektion durch die Inhalte der jeweiligen Sektionen zu ersetzen.

4.1.1 Header

Der Header dient zur eindeutigen Kennzeichnung des Übertragungsformats. Im Falle von OCMF ist in dieser Sektion der String „OCMF“ anzugeben.

4.2 Nutzdatensektion

Die Nutzdatensektion besteht aus einem JSON-Objekt. Darin enthalten sind mehrere Wertzuordnungen bzw. Unter-Objekte. Der Inhalt der Nutzdatensektion darf zwischen Signierung und Verifikation der Signatur nicht verändert werden.

4.2.1 Allgemeine Angaben

Die Angaben in diesem Abschnitt beziehen sich vorzugsweise auf die signaturerzeugende Komponente, welche hier der Einfachheit halber als Gateway bezeichnet wird. Je nach Aufbau ist evtl. die Bezeichnung Messkapsel oder Signierkomponente treffender.

Key:	Typ	Kard.	Beschreibung:
FV	String	1..1	Format-Version: Version des Datenformats in der Darstellung <major>.<minor> Die Versionsangabe wird entsprechend der Version dieses Dokumentes codiert, d.h. entspricht Major 0, Minor 4 Eine Angabe der Revision ist nicht nötig, da dies am Format selbst nichts ändert.
GI	String	0..1	Gateway-Identification: Bezeichner des Herstellers für das System, welches die vorliegenden Daten erzeugt hat (Hersteller, Modell, Variante, etc.).
GS	String	0..1	Gateway-Serial: Seriennummer des o.g. Systems Dieses Feld ist bedingt obligatorisch.
GV	String	0..1	Gateway-Version: Versionsbezeichnung des Herstellers für die SW auf o.g. System Dieses Feld ist optional.

Tabelle 1: Felder zu allgemeinen Angaben

4.2.2 Paginierung

Die Paginierung gibt den Bezug zu einer Transaktion an und ordnet den Gesamtdatensatz im Strom der Ablesungen über die Zähler nachvollziehbar ein. Für die Paginierung stehen mehrere Kontexte zur Verfügung,

jeder Kontext hat einen eigenen Zählerstand. Der jeweilige Paginierungszähler wird monoton aufsteigend mit einem Inkrement von 1 gezählt. D.h. aufeinander folgende Werte des Paginierungszählers zählen fortlaufend im Raum der natürlichen Zahlen.

Der Kontext für Transaktionen ist in jedem Fall zu unterstützen. Der Kontext Fiscal ist optional und kann von der Signaturkomponente optional unterstützt werden, falls auch außerhalb von Transaktionen signierte Ablesungen gewünscht sind.

Key:	Typ:	Kard	Beschreibung:
PG	String	1..1	Paginierung des gesamten Datensatzes, d.h. der Daten, die gemeinsam unter eine Signatur fallen. Format: <Kennzeichen><Zahl> Der String setzt sich aus einem kennzeichnenden Buchstaben für den Kontext und einer Zahl ohne führende Nullen zusammen. Für jeden Kontext existiert ein eigener unabhängiger Paginierungszähler. Folgende Kennzeichen sind definiert: F: Fiscal – Ablesungen unabhängig von Transaktionen (optional) T: Transaction – Ablesungen im Transaktionsbezug (obligatorisch) Der jeweilige Paginierungszähler wird nach jeder Verwendung für einen Datensatz erhöht.

Tabelle 2: Paginierungsfeld

4.2.3 Zähleridentifikation

Die Zähleridentifikation ist optional, wenn die signaturerzeugende Komponente bereits in den allgemeinen Daten identifiziert wurde.

Key:	Typ:	Kard	Beschreibung:
MV	String	0..1	Meter-Vendor: Hersteller-Identifikation des Zählers, Name des Herstellers.
MM	String	0..1	Meter-Model: Modell-Identifikation des Zählers.
MS	String	1..1	Meter-Serial: Seriennummer des Zählers.
MF	String	0..1	Meter-Firmware: Firmware-Version des Zählers.

Tabelle 3: Felder zur Identifikation des Zählers

4.2.4 Benutzerzuordnung

Hierzu gehörige Felder sind genau dann enthalten, wenn Transaktionsbezug besteht. Auch wenn bei Transaktionsbezug noch kein Nutzer zugeordnet werden kann, ist der Abschnitt enthalten. Wenn kein Transaktionsbezug besteht, fehlt diese Gruppe von Feldern.

Key:	Typ:	Kard	Beschreibung:
IS	Boolean	1..1	Identification-Status: Genereller Status zur Benutzerzuordnung: • true: Benutzer erfolgreich zugeordnet • false: Benutzer nicht zugeordnet.
IL	String	0..1	Identification-Level: Aufgeschlüsselter Gesamtstatus der Benutzerzuordnung, dargestellt durch einen Bezeichner aus Tabelle 10
IF	Array of String	0..4	Identification-Flags: Detailaussagen zur Nutzerzuordnung, dargestellt durch einen oder mehrere Bezeichner aus Tabelle 12 bis Tabelle 15. Die Bezeichner werden stets als String-Elemente in einem Array notiert. Auch ein oder kein Element muss als Array notiert werden.
IT	String	1..1	Identification-Type: Typ der Identifikationsdaten, Bezeichner siehe Tabelle 16
ID	String	0..1	Identification-Data: Die eigentlichen Identifikationsdaten entsprechend des Typs aus Tabelle 16, also z.B. eine Hex-Codierte UID entsprechend ISO14443.

Tabelle 4: Felder für die Benutzerzuordnung

4.2.5 Zuordnung des Ladepunktes

Diese optionale Nutzdatensektion bietet eine Möglichkeit zur Identifikation des Ladepunktes. Dies kann eine Alternative zur Identifikation durch Seriennummern darstellen oder zusätzlich genutzt werden.

Key:	Typ:	Kard	Beschreibung:
CT	String	0..1	Charge-Point-Identification-Type: Typ der Angabe zur Identifikation des Ladepunktes, Bezeichner siehe Tabelle 17
CI	String	0..1	Charge-Point-Identification: Identifikations-Angabe für den Ladepunkt.

Tabelle 5: Felder für die Identifikation des Ladepunktes

4.2.6 Ablesungen

Eine oder mehrere Ablesungen können in einem Datensatz abgelegt werden. Jede Ablesung wird in einem Sub-Objekt gekapselt. Diese Objekte werden als Array unter dem Key „RD“ für Reading(s) notiert. Dieses Array enthält somit ein oder mehrere anonyme Objekte.

Jedes dieser Objekte ist aufgebaut, wie in Tabelle 6 beschrieben.

Bei den Ablesungen gilt, dass Felder, die einen identischen Wert zur vorherigen Ablesung haben, ausgelassen werden. Dies gilt allerdings nur innerhalb eines signierten Datensatzes. Das bedeutet konkret, dass beispielsweise nur im ersten Sub-Objekt das Feld „RI“ mit einem Wert (hier dem OBIS-Code) definiert wird und damit für alle weiteren Ablesungen gültig ist. Ebenso kann zum Beispiel das Feld „TX“ im ersten Sub-Objekt als „B“ definiert werden, im zweiten als „C“, im letzten mit „E“, was bedeutet dass die Ablesungen drei bis zur vorletzten den Wert „C“ übernehmen.

Die Felder „RV“, „RI“ , „RU“ und „RT“ können ausgelassen werden, falls nur auf den Eintritt eines Fehlerzustands (Ereignis) des Zählers hingewiesen werden soll.

Die Felder „RI“ und „RU“, bilden eine Gruppe. Felder einer Gruppe sind entweder alle gemeinsam präsent oder werden gemeinsam ausgelassen.

Ein Stromausfall während zeitbasierter Abrechnung stellt ein Fehlerereignis dar.

Key:	Typ:	Kard	Beschreibung:
TM	String	1..1	Time: Angabe zur Systemzeit der Ablesung und Synchronisationszustand. Die Zeit wird nach ISO 8601 mit einer Auflösung von Millisekunden beschrieben. Das Format wird dementsprechend nach folgendem Schema gebildet: <Jahr >-<Monat>- <Tag>T<Stunden>:<Minuten>:<Sekunden>,<Millisekunden><Zeitzone> Dabei wird das Jahr vierstellig dargestellt, Monat, Tag, Stunden, Minuten und Sekunden zweistellig, Millisekunden dreistellig. Die Angabe zur Zeitzone besteht aus einem Vorzeichen und einer vierstelligen Angabe für Stunden und Minuten. Beispiel: 2018-07-24T13:22:04,000+0200 Der Synchronisationszustand besteht aus einem Großbuchstaben als Bezeichner. Dieser wird, getrennt durch ein Leerzeichen, der Zeit hintangestellt. Verfügbare Zustände siehe Tabelle 18.
TX	String	0..1	Transaction: Ablesungsgrund, Bezug der Ablesung zur Transaktion, notiert als Großbuchstabe: B – Beginn der Transaktion C – Charging = während der Ladung (kann optional verwendet werden) X – Exception = Fehler während der Ladung, Transaktion wird fortgesetzt, Zeit und/oder Energie sind ab dieser Ablesung (einschl.) nicht mehr verwertbar. E – Ende der Transaktion, alternativ genauere Codes: L – Ladevorgang wurde lokal beendet R – Ladevorgang wurde aus der Ferne beendet

			A – (Abort) Ladevorgang wurde durch Fehler abgebrochen P – (Power) Ladevorgang wurde durch Stromausfall beendet S – Suspended = Transaktion aktiv, aber gerade keine Ladung (kann optional verwendet werden) T – Tarifwechsel Dieses Feld fehlt, wenn kein Transaktionsbezug vorhanden ist. (Fiscal Metering)
RV	Number	1..1	Reading-Value: Der Wert der Ablesung Hier wird auf das JSON-Datenformat Number zurückgegriffen, dies erlaubt unter anderem eine genaue Auszeichnung der gültigen Nachkommastellen. Die Darstellung darf durch weitere Behandlungsmethoden (z.B. Verarbeitung durch JSON-Parser) jedoch nicht umgeformt werden (Umschreibung der Zahl mit anderem Exponenten, Kürzung von Nachkommastellen, etc.) da damit die Darstellung der physikalischen Größe und damit potentiell die Anzahl der gültigen Stellen verändert würde. Entsprechend der Anwendungsregel wird empfohlen, den Messwert mit zwei Nachkommastellen Genauigkeit darzustellen, falls es sich um kWh handelt.
RI	String	0..1	Reading-Identification: Bezeichner, welche Größe abgelesen wurde, gemäß OBIS-Code.
RU	String	1..1	Reading-Unit: Einheit der Ablesung, z.B. kWh.
RT	String	0..1	Reading-Current-Type: Die vom Zähler gemessene Stromart, z.B. Wechselstrom oder Gleichstrom, gemäß Tabelle 21: Vordefinierte Stromarten . Dieses Feld ist optional. Es ist kein Default-Wert definiert.
EF	String	0..1	Error-Flags: Aussage, welche Größen durch einen Fehler nicht mehr verwertbar zur Abrechnung sind. Jedes Zeichen in diesem String kennzeichnet eine Größe. Folgende Zeichen sind definiert: E – Energie t – Zeit
ST	String	1..1	Status: Zustand des Zählers zum Zeitpunkt der Ablesung. Notiert als Kürzel gemäß Tabelle 9.

Tabelle 6: Felder eines Ablesungs-Objekts

4.2.7 Erweiterungspunkte

In der Nutzdatensektion ist es für verschiedene Hersteller eventuell nötig, dass diese eigene Daten hinzufügen können. Dazu werden im Namensraum des JSON-Objekts folgende Erweiterungspunkte zur freien Benutzung in Abhängigkeit vom Hersteller reserviert:

- Reserviert sind alle Namen auf oberster Ebene im JSON-Objekt der Nutzdatensektion, welche mit folgenden Buchstaben beginnen:
  - U
  - V
  - W
  - X
  - Y
  - Z
- Da in der Nutzdatensektion nachträglich keine Veränderungen oder Hinzufügungen erfolgen können (sic), sind keine entsprechenden Erweiterungspunkte definiert. Siehe unten, Sektion 1.1.

4.3 Signatursektion

Die Signatursektion besteht ebenfalls aus einem eigenständigen JSON-Objekt. Darin enthalten sind mehrere Wertzuordnungen bezüglich der Signaturdaten.

4.3.1 Signaturdaten

Key:	Typ:	Kard	Beschreibung:

SA	String	0..1	Signature-Algorithm: Wählt den zur Signaturbildung verwendeten Algorithmus aus. Dies umfasst den Signatur-Algorithmus, dessen Parameter und den Hash-Algorithmus der auf den zu signierenden Daten angewendet wird. Diese Angabe ist optional. Wird sie ausgelassen, ist der Default-Wert gültig. Tabelle 22 zeigt, welche Werte möglich sind.
SE	String	0..1	Signature-Encoding: Bezeichnet, wie die Signaturdaten kodiert sind, um im JSON-String abgelegt werden zu können. Diese Angabe ist optional. Wird sie ausgelassen, ist der Default-Wert gültig. Folgende Werte sind möglich: hex – Die Signaturdaten sind im JSON-String hexadezimal codiert dargestellt (Default) base64 – Die Signaturdaten sind entsprechend base64 abgelegt
SM	String	0..1	Signature-Mime-Type: Bezeichnet, wie die Signaturdaten zu interpretieren sind. Diese Angabe ist optional. Wird sie ausgelassen, ist der Default-Wert gültig. Folgende Werte sind möglich: application/x-der – Ablage als ASN.1-codierte .der-Datei (Default)
SD	String	1..1	Signature-Data: Die eigentlichen Signaturdaten entsprechend obiger Formatangaben.

Tabelle 7: Signaturdatenfelder der Signatursektion

4.3.2 Erweiterungspunkte

Analog zu den Erweiterungspunkten in Sektion 1.1, sind eventuell auch ungeschützte Erweiterungspunkte wünschenswert. Folgende sind dazu reserviert:

- Für die Verwendung durch Hersteller sind alle Namen auf oberster Ebene im JSON-Objekt der Signaturdatensektion reserviert, welche mit folgenden Buchstaben beginnen:
  - U
  - V
  - W
  - X
  - Y
  - Z
- Für die Verwendung durch Komponenten in der nachgelagerten Verarbeitung von Daten sind entsprechend folgende Anfangsbuchstaben reserviert:
  - A
  - B
  - C
  - D
  - E
  - F

4.4 Bezug von Seriennummern, Ladepunkt und Public-Key

Zum Aufbau eines Bezugs zwischen dem zur Signaturprüfung zu benutzenden Public-Key und des Ladepunktes dienen eine oder mehrere Seriennummern, alternativ kann auch eine direkte Identifikation des Ladepunktes per ID erfolgen.

Die Public-Keys zum Ladepunkt müssen auf anderem Wege als diesem Protokoll (Out-of-Band) an die Prüfkomponente in Verbindung mit der verwendeten ID oder den Seriennummern übermittelt werden, z.B. über ein zentrales Register.

Um eine Signaturkomponente eindeutig zu identifizieren werden in diesem Protokoll folgende Möglichkeiten definiert:

- Ist die Signaturkomponente dem Energiezähler zuzurechnen, reicht dessen Seriennummer.
- Ist die Signaturkomponente selbst das Gateway und nur ein Ladepunkt wird bedient, reicht die Seriennummer des Gateways oder des Energiezählers.
- Ist ein Gateway für mehr als einen Ladepunkt zuständig, wird die Kombination aus den Seriennummern von Gateway und Energiezähler zur eindeutigen Identifikation herangezogen.
- Alternativ kann eine direkte Identifikation des Ladepunktes erfolgen.

Da diese Bedingungen vom Aufbau der Ladestation abhängen, sind die entsprechenden Seriennummernfelder weiter unten als bedingt obligatorisch gekennzeichnet.

4.5 Beispiel

Beispiele zu den resultierenden JSON-basierten Formaten:

```
OCMF|{
    "FV": "1.0",
    "GI": "ABL SBC-301",
    "GS": "808829900001",
    "GV": "1.4p3",

    "PG": "T12345",

    "MV": "Phoenix Contact",
    "MM": "EEM-350-D-MCB",
    "MS": "BQ27400330016",
    "MF": "1.0",

    "IS": true,
    "IL": "VERIFIED",
    "IF": [ "RFID_PLAIN", "OCPP_RS_TLS" ],
    "IT": "ISO14443",
    "ID": "1F2D3A4F5506C7",

    "RD": [
        {
            "TM": "2018-07-24T13:22:04,000+0200 S",
            "TX": "B",
            "RV": 2935.6,
            "RI": "1-b:1.8.0",
            "RU": "kWh",
            "RT": "AC",
            "EF": "",
            "ST": "G"
        }
    ]
}|{
     "SD": "887FABF407AC82782EEFFF2220C2F856AEB0BC22364BBCC6B55761911ED651D1A922BADA88818C9
671AFEE7094D7F536"
}
```

Abbildung 1: Beispieldatei Darstellungsformat

5 Weitere Datenfelder

5.1 Zähleridentifikation

5.1.1 Hersteller und Modell des Zählers

Die folgende Tabelle stellt die vordefinierten Werte für die Felder Hersteller und Modell des Zählers dar.

Hersteller:	Modell:
Phoenix Contact	EEM-350-D-MCB
Inepro Metering	Pro 1 Mod
	Pro 380 Mod
Carlo Gavazzi	EM340-DIN.AV2.3.X.S1.PF
	EM111-DIN.AV8.1.X.S1.PF
	EM210-72D.MV5.3.X.OS.X
NZR	EcoCount S 85

Tabelle 8: Werte für Hersteller/Modell des Zählers

5.1.2 Zustand / Fehler des Zählers

Dieses Feld kann gleichzeitig einen Wert annehmen.

Kürzel:	Bezeichner:	Beschreibung:
N	NOT_PRESENT	Der Zähler ist nicht vorhanden/gefunden worden
G	OK	Zähler in Ordnung (Good)
T	TIMEOUT	Zeitüberschreitung beim Versuch, den Zähler anzusteuern
D	DISCONNECTED	Zähler wurde von der Signaturkomponente getrennt
R	NOT_FOUND	Zähler nicht mehr gefunden (ist bereits vorher mind. einmal gefunden worden) (Removed)
M	MANIPULATED	Manipulation erkannt
X	EXCHANGED	Zähler getauscht (Seriennummer stimmt nicht mehr mit der zuletzt bekannten überein.)
I	INCOMPATIBLE	Zähler-Version oder dessen API nicht mit Signaturkomponente kompatibel
O	OUT_OF_RANGE	Gelesener Wert außerhalb des Wertebereichs
S	SUBSTITUTE	Es wurde ein Ersatzwert gebildet
E	OTHER_ERROR	Anderer, nicht näher bekannter Fehler
F	READ_ERROR	Zählerregister nicht korrekt ausgelesen; Wert der Auslesung ist nicht gültig

Tabelle 9: Zustand/Fehler des Zählers

5.2 Benutzerzuordnung

5.2.1 Status / Fehler

Dieses Feld kann gleichzeitig nur einen Wert annehmen. Zur Übersichtlichkeit sind die Werte in mehrere Gruppen eingeteilt, die im Format nicht ausgezeichnet werden.

Gruppe:	Status/Fehler:	Beschreibung:
nicht verfügbar	-	Das Feld ist nicht angegeben.
Status ohne Zuordnung	NONE	Es liegt keine Nutzerzuordnung vor. Die weiteren Daten zur Nutzerzuordnung haben keine Aussagekraft.

Status mit Zuordnung	HEARSAY	Die Zuordnung ist ungesichert; also z.B. durch Auslesen einer RFID-UID.
	TRUSTED	Die Zuordnung kann in gewissem Maße vertraut werden, es gibt jedoch keine absolute Verlässlichkeit. Beispiel: Authorisierung durch Backend
	VERIFIED	Die Zuordnung wurde durch die Signaturkomponente und spezielle Maßnahmen verifiziert.
	CERTIFIED	Die Zuordnung wurde durch die Signaturkomponente verifiziert mit Hilfe einer kryptographischen Signatur die die Zuordnung zertifiziert.
	SECURE	Die Zuordnung wurde durch eine sicheres Merkmal hergestellt (z.B. sichere RFID-Karte, ISO15118 mit Plug-and-Charge, etc.)
Fehler	MISMATCH	Fehler; UIDs passen nicht zueinander.
	INVALID	Fehler; Zertifikat nicht korrekt (Prüfung negativ).
	OUTDATED	Fehler; referenziertes Trust-Zertifikat abgelaufen.
	UNKNOWN	Zertifikat konnte nicht erfolgreich geprüft werden (kein passendes Trust-Zertifikat gefunden).

Tabelle 10: Status- und Fehlerzustände der Nutzerzuordnung

5.2.2 Details

Dieses Feld setzt sich aus vier Sub-Feldern zusammen und gibt Details zum Zustandekommen des Status der Benutzerzuordnung an.

Tabelle 11 beschreibt die Aufteilung in Subfelder, die darauf folgenden Tabellen beschreiben die jeweils gültigen Werte.

		Inhalt:	Details siehe:
		Zuordnungsanteil RFID	Tabelle 12
		Zuordnungsanteil OCPP	Tabelle 13
		Zuordnungsanteil ISO 15118	Tabelle 14
		Zuordnungsanteil PLMN	Tabelle 15

Tabelle 11: Aufteilung Details Nutzerzuordnung

	Bezeichner:	Beschreibung:
	RFID_NONE	keine Zuordnung per RFID
	RFID_PLAIN	Zuordnung über externen RFID-Kartenleser
	RFID_RELATED	Zuordnung über geschützten RFID-Kartenleser
	RFID_PSK	Ein vorher bekannter gemeinsamer Schlüssel (Pre-shared key) wurde genutzt, z.B. mit einer abgesicherten RFID-Karte.

Tabelle 12: Zuordnungsanteil RFID

	Bezeichner:	Beschreibung:
	OCPP_NONE	keine Nutzerzuordnung durch OCPP
	OCPP_RS	Zuordnung durch OCPP-RemoteStart-Methode
	OCPP_AUTH	Zuordnung durch OCPP-Authorize-Methode
	OCPP_RS_TLS	Transport-Layer-Security wurde zur Übertragung der Zuordnung per OCPP-RemoteStart-Methode benutzt.

	OCPP_AUTH_TLS	Transport-Layer-Security wurde zur Übertragung der Zuordnung per OCPP-Authorize-Methode benutzt.
	OCPP_CACHE	Zuordnung durch Authorization-Cache von OCPP
	OCPP_WHITELIST	Zuordnung durch White-List von OCPP
	OCPP_CERTIFIED	Ein Zertifikat des Backends welches die Nutzerzuordnung zertifiziert wurde eingesetzt.

Tabelle 13: Zuordnungsanteil OCPP

	Bezeichner:	Beschreibung:
	ISO15118_NONE	keine Nutzerzuordnung durch ISO15118
	ISO15118_PNC	Plug & Charge wurde benutzt

Tabelle 14: Zuordnungsanteil ISO 15118

	Bezeichner:	Beschreibung:
	PLMN_NONE	keine Zuordnung
	PLMN_RING	Anruf
	PLMN_SMS	Kurznachricht

Tabelle 15: Zuordnungsanteil PLMN

5.2.3 Typ

Tabelle 16 beschreibt die möglichen Typen von Nutzerzuordnungen, welche in dem Feld Typ als vorzeichenlose Integer verwendet werden können. Diese Zuordnungen orientieren sich an den Vorgaben aus OCPP. Allerdings sieht OCPP derzeit (Version 1.5) nur 20 Zeichen für das Identifikationsmerkmal vor. Entsprechend der maximalen Länge einer IBAN (34 Zeichen) wurden für den Datenbereichs hier allerdings Zuordnungen bis zu 40 Byte Länge vorgesehen.

Kürzel:	Bezeichner:	Beschreibung:
0	NONE	Keine Zuordnung verfügbar
1	DENIED	Zuordnung momentan nicht abrufbar (wg. Zwei-Faktor-Autorisierung)
2	UNDEFINED	Typ nicht näher benannt
10	ISO14443	UID einer RFID-Karte entsprechend ISO 14443. Dargestellt als 4 oder 7 Bytes in hexadezimaler Schreibweise.
11	ISO15693	UID einer RFID-Karte entsprechend ISO 15693. Dargestellt als 8 Bytes in hexadezimaler Schreibweise.
20	EMAID	Electro-Mobility-Account-ID entsprechend ISO/IEC 15118 (String mit der Länge 14 oder 15)
21	EVCCID	ID eines Elektrofahrzeugs entsprechend ISO/IEC 15118 (Maximallänge 6 Zeichen)
30	EVCOID	EV-Contract-ID entsprechend DIN 91286.
40	ISO7812	Identification-Card-Format entsprechend ISO/IEC 7812 (Kredit- und Bankkarten, etc.)
50	CARD_TXN_NR	Kartentransaktionsnummer (CardTxNbr) für eine Zahlung mit Kredit- oder Bankkarte, die in einem Terminal am Ladepunkt benutzt wird.
60	CENTRAL	Zentral generierte ID. Kein genaues Format definiert, kann z.B. eine UUID sein. (OCPP 2.0)
61	CENTRAL_1	Zentral generierte ID, z.B. durch Start per SMS. Kein genaues Format definiert. (bis OCPP 1.6)

62	CENTRAL_2	Zentral generierte ID, z.B. durch Start durch Operator. Kein genaues Format definiert. (bis OCPP 1.6)
70	LOCAL	Lokal generierte ID. Kein genaues Format definiert, kann z.B. eine UUID sein. (OCPP 2.0)
71	LOCAL_1	Lokal generierte ID, z.B. intern durch den Ladepunkt erzeugte ID. Kein genaues Format definiert. (bis OCPP 1.6)
72	LOCAL_2	Lokal generierte ID, für sonstige Fälle. Kein genaues Format definiert. (bis OCPP 1.6)
80	PHONE_NUMBER	Internationale Telefonnummer mit führendem „+".
90	KEY_CODE	User-bezogener, privater Key-Code. Kein genaues Format definiert.

Tabelle 16: Typen der Nutzeridentifikation

5.3 Zuordnung des Ladepunktes

Bezeichner:	Beschreibung:
EVSEID	EVSE-ID
CBIDC	Charge-Box-ID und Connector-ID nach OCPP, als Feldtrenner wird ein Leerzeichen benutzt, z.B. „STEVE_01 1".

Tabelle 17: Typen der Ladepunktzuordnung

5.4 Status der Zeit

Bezeichner:	Beschreibung:
U	unbekannt, unsynchronisiert
I	informativ (Info-Uhr)
S	synchronisiert
R	relative Zeitabrechnung mit einem eichrechtlich akkuraten Zeitgeber, basierend auf einer Info-Uhr. D.h. zu Beginn des Ladevorgangs war eine Zeit mit informativer Qualität vorhanden. Während des Ladevorgangs wurde die Zeit (Dauer) entsprechend der eichrechtlichen Anforderungen erfasst.

Tabelle 18: Status der Zeit

5.5 Beschaffenheit des Messwerts

5.5.1 OBIS-Codes

Um ein Format in Version 1.0 zu definieren, wurden hier verschiedene mögliche OBIS-Codes aufgelistet. Im Nachgang kann noch geklärt werden, welche davon zu verwenden sind. Die Formate aus der Vorgängerversion 0.3 wurden beibehalten.

TODOs:

- Kommentar(e) zu den OBIS-Codes noch analysieren. Dies führt evtl. zu veränderten OBIS-Codes.

OBIS-Code:	Beschreibung:
1-b:1.8.e	Bezug von elektrischer Energie (Wirkarbeit) aus dem Stromnetz (des Ladepunktbetreibers) an den Kunden.
1-b:2.8.e	Lieferung von elektrischer Energie (Wirkarbeit) von dem Kunden an das Stromnetz (des Ladepunktbetreibers).
1-b:1.8.0	Bezug von elektrischer Energie (Wirkarbeit) aus dem Stromnetz (des Ladepunktbetreibers) an den Kunden.
1-b:2.8.0	Lieferung von elektrischer Energie (Wirkarbeit) von dem Kunden an das Stromnetz (des Ladepunktbetreibers).

1-0:1.8.0	Bezug von elektrischer Energie (Wirkarbeit) aus dem Stromnetz (des Ladepunktbetreibers) an den Kunden.
1-0:2.8.0	Lieferung von elektrischer Energie (Wirkarbeit) von dem Kunden an das Stromnetz (des Ladepunktbetreibers).

Tabelle 19: Vordefinierte OBIS-Codes

5.5.2 Einheiten

Einheit:
kWh
Wh

Tabelle 20: Vordefinierte Einheiten

5.5.3 Stromart

Bezeichner:	Beschreibung:
AC	Wechselstrommessung
DC	Gleichstrommessung

Tabelle 21: Vordefinierte Stromarten

5.6 Kryptographie

5.6.1 Signatur-Methoden

Bezeichner:	Signatur-Algorithmus:	Kurvenname und ggf. Synonyme:	Schlüssel-länge:	Hash-Algorithmus:	Block-Länge:
ECDSA-secp192k1-SHA256	ECDSA	secp192k1	192 bit	SHA-256	48
ECDSA-secp256k1-SHA256	ECDSA	secp256k1	256 bit	SHA-256	64
ECDSA-secp192r1-SHA256	ECDSA	secp192r1	192 bit	SHA-256	48
ECDSA-secp256r1-SHA256 (Default in OCMF Version 0.4)	ECDSA	secp256r1, NIST P-256, ANSI X9.62 elliptic curve prime256v1	256 bit	SHA-256	64
ECDSA-brainpool256r1-SHA256	ECDSA	brainpool256r1	256 bit	SHA-256	64
ECDSA-secp384r1-SHA256	ECDSA	secp384r1, NIST P-384, ANSI X9.62 elliptic curve prime384v1	384 bit	SHA-256	96
ECDSA-brainpool384r1-SHA256	ECDSA	brainpool384r1	384 bit	SHA-256	96

Tabelle 22: Vordefinierte Signatur-Algorithmen und ihre Parameter

5.6.2 Public-Key-Typen

Bezeichner:	Signatur-Algorithmus:	Kurve:	Schlüssel-länge[1]:	Block-Länge:
ECDSA-secp192k1	ECDSA	secp192k1	192 bit	48

[1] Schlüssellänge des entsprechenden Private-Key

ECDSA-secp256k1	ECDSA	secp256k1	256 bit	64
ECDSA-secp192r1	ECDSA	secp192r1	192 bit	48
ECDSA-secp256r1 (Default in OCMF Version 0.4)	ECDSA	secp256r1	256 bit	64
ECDSA-brainpool256r1	ECDSA	brainpool256r1	256 bit	64
ECDSA-secp384r1	ECDSA	secp384r1	384 bit	96
ECDSA-brainpool384r1	ECDSA	brainpool384r1	384 bit	96

Tabelle 23: Vordefinierte Schlüsseltypen