ITK ENGINEERING

SERVICE OVERVIEW

CHARGE ENABLER: OSS for securing ISO 15118 compliant charging

The Charge Enabler as an Open-Source tool offers a new ISO 15118-compliant basis for testing market-specific requirements for seamless communication between charging station and vehicle.

Motivation

With increasing sustainability awareness and the expansion of renewable energies, interest in electromobility is also growing. However, the increasing number of electric vehicles in the European market also means an increased demand for suitable charging infrastructure. This infrastructure is still inadequately developed, and the associated potential challenges of route planning are a reason not to use electric vehicles. Numerous companies have rediscovered this market for themselves and want to position themselves. However, new developments take more time than planned, because errors in software development are only noticed late or often remain undetected until release and are only detected during operation in the form of charging interruptions. Even experienced companies in the field of electromobility, such as OEMs in the automotive sector and manufacturers of charging infrastructures, are repeatedly faced with the challenge of detecting charging interruptions at an early stage with a suitable test setup, to minimize and ultimately optimizing their developments in their entirety.

In addition to a potential loss of image, this also means considerable additional effort and costs for the development. On the one hand, the cause must be found and, if necessary, solved on the software side, on the other hand, the software must be adapted during operation.

The Charge Enabler was developed to meet these challenges. It forms an Open Source testing framework for ISO 15118 standard-compliant charging processes in early development phases for software-side validation based on compliance tests to detect software errors at an early stage, to optimize development and thus reduce charging interruptions during operation. In this way, the development can be accelerated, late detection of errors minimized and thus development and maintenance costs reduced.

Electric charging

In Europe, there are various options for the electrical energy supply of electric vehicles. A distinction is made here between different charging types: from Conductive charging with alternating current (AC charging), via charging with direct current (DC), up to inductive charging. Depending on the charging method, the actual charger and the charging line are located in the vehicle itself, or in a charging station with different communication interfaces.

In addition, the charging processes are differentiated by their charging power. Up to 22 kW is called „normal charging“, between 22 kW and 50kW as „fast charging“ and above 50 kW as „high-performance charging“. AC charging is intended for normal and fast charging, while DC charging is also possible in the area of high-power charging, equivalent to High-Power-Charging (HPC).

These different charging modes and processes result in the following charging modes for wired charging [DIN EN IEC 61851-1]:

Charging mode 1 / mode 1 describes charging with alternating current at a earthing contact socket or an industrial socket with one or three phases. This charging mode does not require a defined communication interface between the vehicle and the charging infrastructure.

Charging mode 2 / mode 2 also describes charging with alternating current. The difference to mode 1, however, is a charging cable integrated in the vehicle, which includes control and protection. A pilot signal is used to initiate information exchange and monitoring.

Charging mode 3 / mode 3 describes a third variant of charging with alternating current. One or three phases are also used here, but a permanently installed charging station is required to which the charging line is connected. Safety features with residual current protection are integrated in the overall installation. All communication takes place via the charging line.

Charging mode 4 / mode 4 again describes charging with direct current. This charging process requires permanently installed charging stations that have a charging line and the actual charger. As in mode 3, all communication takes place via the charging line.

Modes 2, 3 and 4 are based on the so-called „low level" communication according to IEC 61851-1, through which basic information is exchanged. Additional communication in accordance with ISO 15118 is possible in charging mode 3.

In development, this multitude of charging modes and the associated variations must be taken into account in order to avoid malfunctions during operation. Not least because, in addition to the dissatisfaction of users, it is also about minimizing damage. In most cases, the charging interruptions that occur during operation, which can be experienced by the customer, are due to communication problems between the vehicle and the charging infrastructure, which are mainly due to potential software weaknesses and inadequate implementation of the communication standard. The challenge for development is therefore to master the challenge of standard-compliant communication. In Europe, ISO 15118 is generally used as the standard for this purpose.

ISO 15118 as a framework

The communication standard ISO 15118 - Road Vehicles - Vehicle to Grid Communication Interface maps the necessary communication, identification, authentication, and authorization processes between electric vehicles and charging infrastructure.

The bi-directional communication according to this international standard allows the exchange of data between the electric vehicle and the charging station, such as the State of Charge of the battery (SoC). Energy requirements and charging performance ensure effective charging processes as well as the development of convenient billing systems. It thus guarantees a future-proof and simple charging infrastructure. This open and standardized interface should be continuously developed and expanded with the increasing spread of electromobility in general. In order to check the communication during a charging process for its compliance with standard, compliance tests are available, which are at least partially provided in ISO 15118.

Compliance tests

As mentioned earlier, the high number of charging interruptions is mainly due to software errors in communication. For this reason, it makes sense to start with validation and quality assurance in the early development phases. In particular, compliance tests should be the focus here.

The term compliance in the field of software assurance states: „The ability of a software product to meet application-specific standards or agreements, or statutory and similar regulations [ISO 9126]." with regard to the following quality features: Functionality, reliability, usability, efficiency, changeability, or transferability. In short: testing for standard compatibility. For this purpose, functional black box test cases are usually used, which verify the communication against the underlying protocol. These test cases are derived from the specification and adjust the implementation and internal structure of the system to be tested (black box).

To ensure interoperability between the vehicle (EVCC - Electric Vehicle Communication Controller) and the charging infrastructure (SECC - Supply Equipment Communication Controller), this charging standard-specific verification should be carried out. The standardized conformance tests contained in ISO 15118-4 ensure compliance of network and application protocols in accordance with ISO 15118-2. These test case implementations are in a domain-specific, formal programming language for testing communication-based systems - called TTCN-3 Testing and Test Control Notation"- and can be interpreted, for example, by the free Eclipse Titan Compiler. It provides an integrated development and execution environment for modular and logical test case description.

The Charge Enabler offers the option of stimulating the component to be tested and performs the standardized compliance tests with subsequent evaluation.

Charge Enabler: Making complexity manageable

The Charge Enabler has therefore set itself the goal of enabling the early detection of software errors in communication and thus contributing to quality assurance in order to save costs. There are already several test frameworks on the market for securing charging processes in electromobility. But on the one hand these are very expensive and on the other hand the target hardware is required. The Charge Enabler offers a modular Open Source test framework, which is already used in early development stages and does not require any hardware. This ensures compliance on the software side and independently of possible hardware supply bottlenecks.

The customer-specific SECC or EVCC software can be integrated with a few adaptations and checked for compliance.

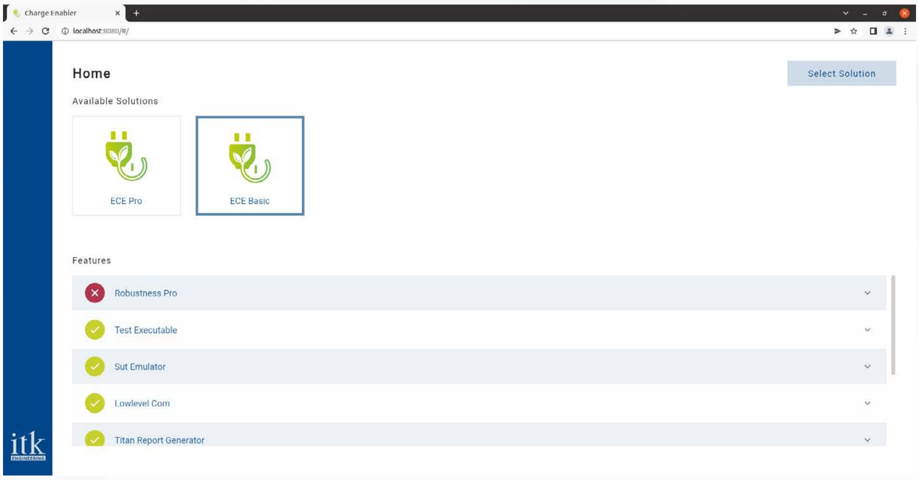

Figure 1 Home Scrreen

Various test suites from ISO 15118 are available for this purpose. The test results provide information about non-standard implementation of the charging standard. Pure software testing makes troubleshooting easier, as the effects of the hardware are excluded. Via a user-friendly web interface, the user has the option of operating the Charge Enabler, parameterizing test suites for compliance tests on an application-based basis and, if required, saving them for later executions.

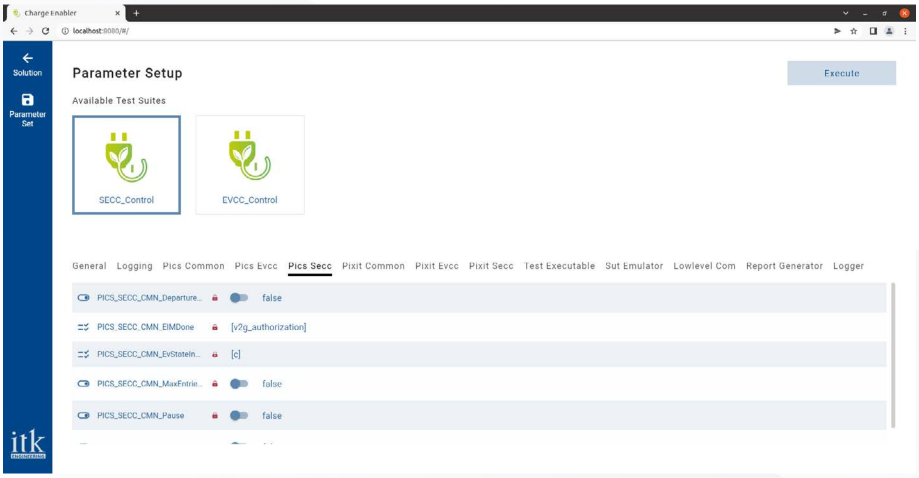

Figure 2 Parameter Setup Screen

Based on the set parameters, the test cases are executed fully automatically. During execution, the test case status is updated. The tester always has the option of pausing, continuing or canceling the test execution.

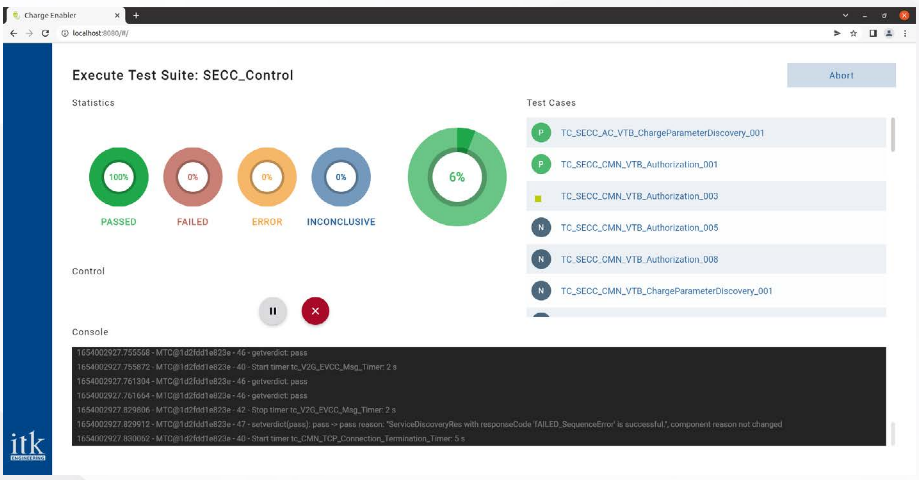

Figure 3 Test Execution Screen

When the test suite is finished, the test results of the test run are displayed. The final result is visualized, and the percentages divided into „PASSED“, „FAILED“, „ERROR“ and „IN CONCLUSIVE“, is indicated.

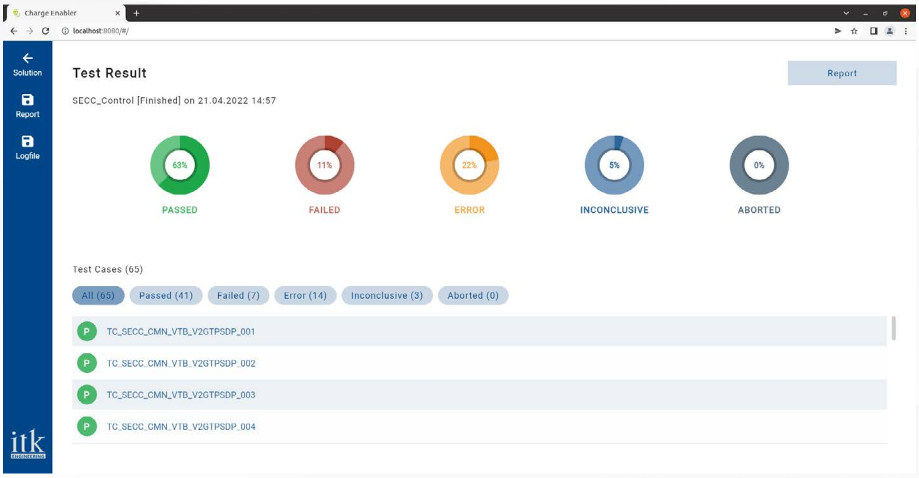

Figure 4 Test Result Screen

The test runs are stored and can subsequently be executed again with the additional parameters.

Figure 5 Test Run Screen

Due to the modular structure of the Charge Enabler and the technologies used in it, it is relatively easy to extend the already existing functionalities with various features, such as customer-specific screens or evaluations.

The architecture

As described above, the Charge Enabler impresses with its modular structure and few dependencies within the software architecture. The gray portions identify used open source software (OSS) within the Charge Enabler, the white portions identify the OSS developed by ITK.

Figure 6 Software Layer

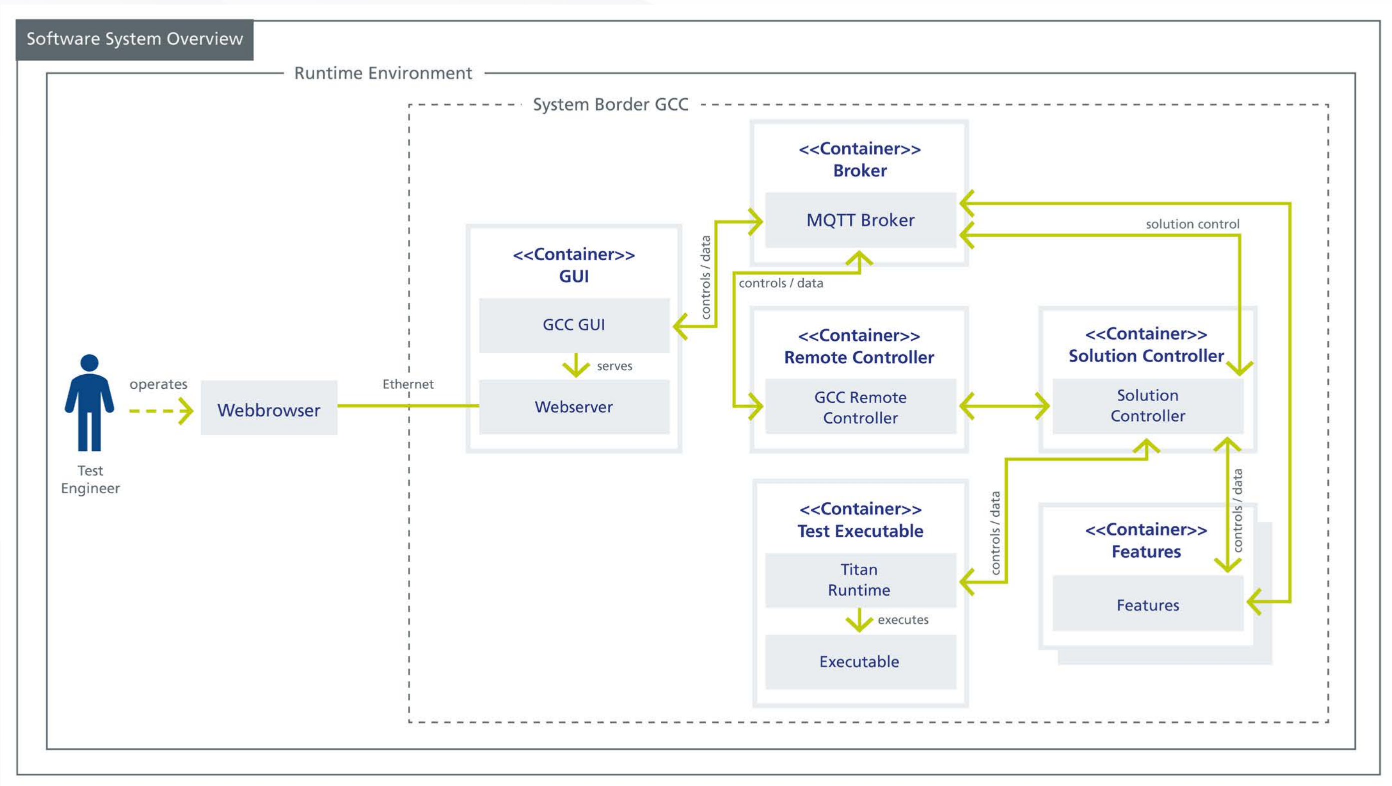

Figure 7 Container dependencies

The functionalities of the software components are largely separated in terms of content:

- GUI: The GUI (as a Web application) provides the user with the possibility of preparing, setting up and executing the compliance tests.
- Remote Controller: The Remote Controller is the most important component of the software system and controls all other components. It connects the user interface with the actual software solution and provides all necessary information.
- Solution Controller: The Solution Controller controls all software components used for the actual software solution and is in particular responsible for the pre- and post-processing of these.
- Test Executable: The Test Executable contains and controls all necessary functionalities for test execution.
- Featues:

▪ Low Level: The low level communication provides the IEC 61851-1 simulator. This is the virtualization of the cable and its properties. The Charge Enabler provides an API so that the System Under Test can address this cable.

▪ SUT Emulator: The SUT Emulator includes the „System Under Test“, emulated by the Rise-V2G, a certified software component that conforms to the communication protocol of ISO 15118 (both for SECC and EVCC).

▪ Report Generator: The Report Generator compiles test execution information in JUnit xml format. The report contains information such as the list of executed and aborted test cases, the test result of the individual test cases, the duration of the execution and potential errors in the test case execution.

▪ Logger: The Logger combines information from the individual software components during the test execution of the actual software solution and stores it as a separate file together with the test results.

- MQTT Blocker: The MQTT broker is the basis of communication via MQTT.

The individual software components are organized in Docker containers and communicate via MQTT. These technologies and their peculiarities are described in more detail below.

Docker Container

Docker Containers are primarily suitable for isolating software applications. A Docker image forms the basis for the containers in the form of a disk image. The actual code and all dependencies are packaged here and enable a fast and reliable transfer to different environments. Thus, the containerized software is independent of the infrastructure used and can be used under any operating systems such as Linux or Windows. For the use of Docker Containers in the Charge Enabler context, platform independence is not directly given. Docker does not currently support Internet Protocol version 6 IPv6 on Windows, which is used according to ISO 15118 for direct communication between the SUT and the test executable. However, this independency can be established via the detour of a virtual Linux machine under Windows.

The Charge Enabler has the following container dependencies:

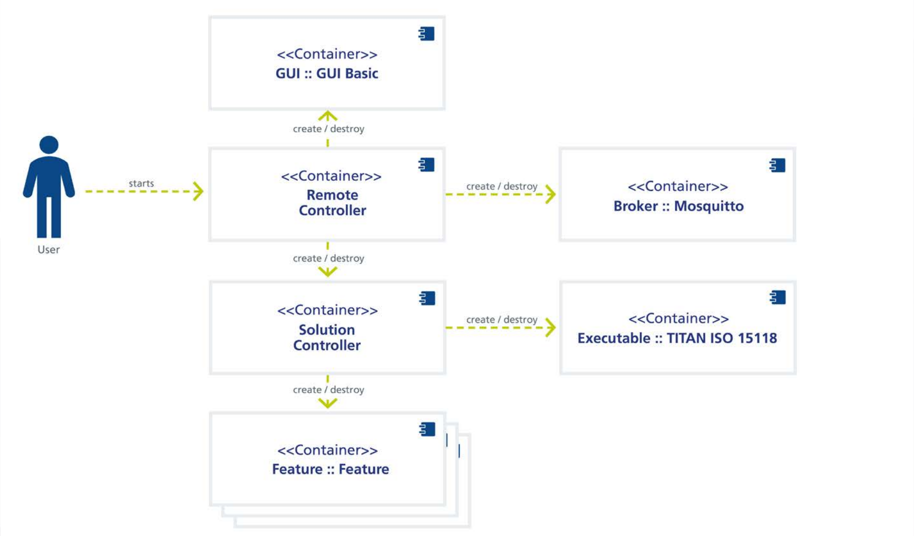

Figure 8 Container dependencies

It is striking that the Remote Controller is the only component that controls an interaction between the user and the software. It also becomes clear that the Solution Controller in turn takes over the control of the actual software solution. This independent architecture keeps costs down.

MQTT

The Message Queuing Telemetry Transport (MQTT) is an open network for Machine-to-Machine Communication (M2M) based on a very simple Client-Server protocol. MQTT is particularly suitable for systems with low bandwidth or unreliable networks, as the protocol reduces resource utilization and ensures stable message transmission. This ensures the supply in general and enables the stable transmission of messages between devices. The server, also called MQTT Broker, plays a key role here. The entire communication of all clients (communication partners) takes place via this Broker. The clients can subscribe to topics in advance and the Broker will only send the published messages to the actual subscribers.

The communication of the Charge Enabler via MQTT corresponds to the following structure:

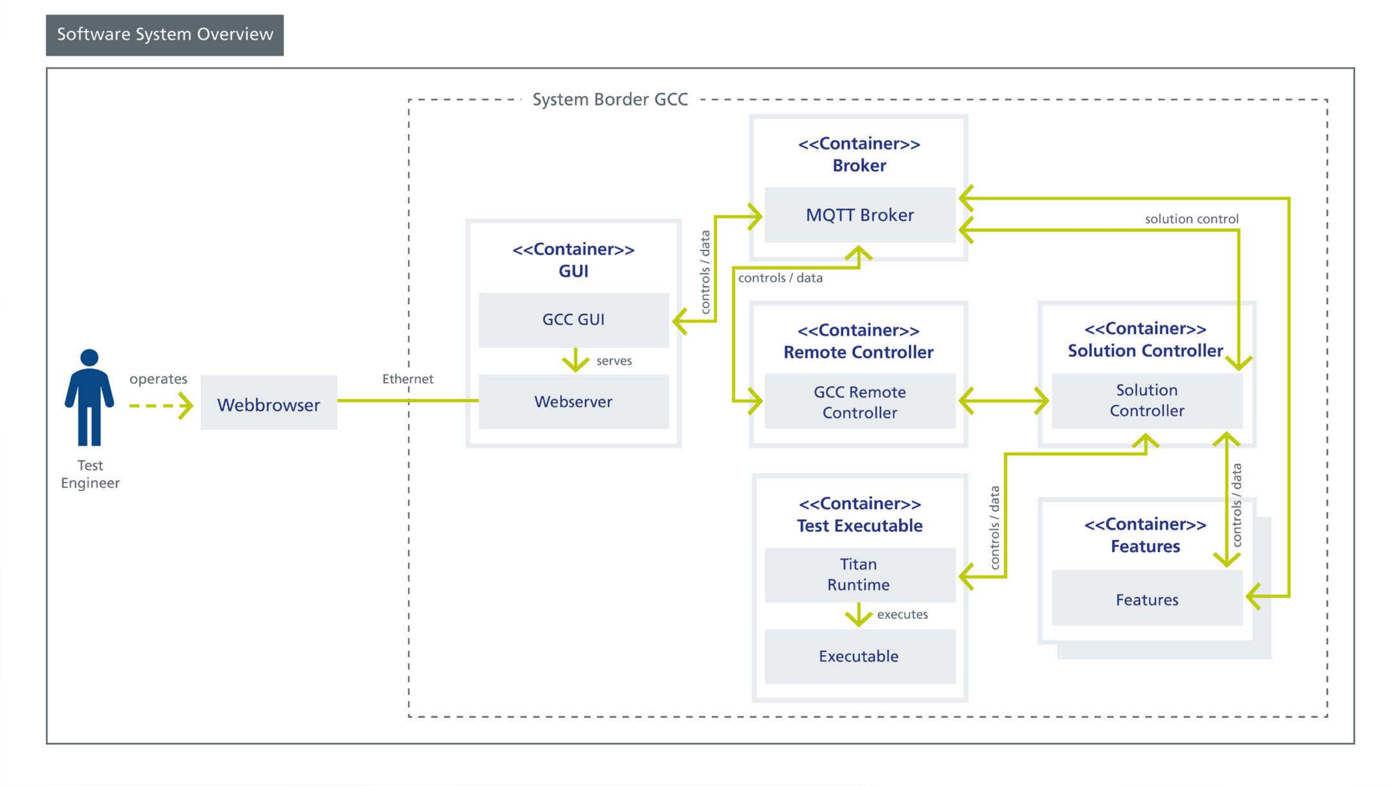

Figure 9 MQTT Communication

MQTT is not considered particularly secure, as the publishers and subscribers usually communicate unencrypted and without authentication. This makes it possible for potential attackers not only to read along, but also to manipulate sequences. With the correct configuration of the Broker using authentication via user name and password and the use of TLS encryption, this security problem can be avoided.

In the Charge Enabler, however, the MQTT data stream is not encrypted because the software is operated on a local computer, in the VPN network or behind a proxy with encryption. Thus, an additional security of the MQTT communication is not necessary.

Rise-V2G

The Rise-V2G is part of the System Test (SUT). The SUT Emulator can be both the vehicle and the charging infrastructure. Rise-V2G stands for „Reference Implementation Supporting the Evolution of the Vehicle-2-Grid Communication Interface ISO 15118“. It is available as an ISO 15118 compliant Open Source software component. This is not an ITK in-house development: https://github.com/SwitchEV/RISE-V2G

Summary

Through the use of these innovative technologies and their independence, the Charge Enabler succeeds in making the complexity of compliance with charging manageable and allows the user a simple operation of the software verification for compliance with standard. New customer-specific requirements for new features can also be reliably implemented due to the technologies used.

Overview

As an Open Source testing framework, the Charge Enabler offers an opportunity for early and cost-effective software testing for compliance with standard. Due to its modular structure and the simple communication structure, almost any extensions can be connected to the Charge Enabler basic software.

Planned enhancements include, for example, the Pro-Reporting feature for detailed test results, an extension of the test suites to customer-specific test cases or robustness test cases and the use of other charging protocols, such as CHAdeMO and OCPP.

But the later use of specific customer hardware can also be implemented by the architecture with a few modifications, so that the tool can also be used after software testing in the HiL (hardware-in-the-loop) area. Thus, identical test cases can be reproducibly executed and analyzed in different test environments. Interactions between software and hardware can thus be analyzed and understood with minimal effort.

With the internal further development of the free basic version of the Charge Enabler, new paid Pro features are gradually being developed. Depending on the application, these features can be integrated into the Charge Enabler Basis as desired. This results in a test system tailored to the needs of the customer. Since all features are developed under the requirement of backward compatibility, a later retrofitting and upgrading of the basic tool is not a major challenge.

Feel free to contact us! At any time, we are available to advise you and support you with a customized solution for your complex problems! In addition, we offer individual workshops and customized training courses on request.

https://www.itk-engineering.de/stories/charge-enabler/

itk ENGINEERING

ITK Engineering GmbH
Headquarters Rülzheim
Im Speyerer Tal 6
76761 Rülzheim

Tel.: + 49 (0)7272 7703-0
Fax: + 49 (0)7272 7703-100
info@itk-engineering.de

www.itk-engineering.de
www.itk-karriere.de

Follow us on: