

USENIX

THE ADVANCED COMPUTING
SYSTEMS ASSOCIATION

Charge It to My Neighbor: A Relay Attack on ISO 15118 Plug and Charge Payment

Jakob Löw, Vishwa Vasu, Thomas Hutzelmann,
and Hans-Joachim Hof, *Technische Hochschule Ingolstadt*

https://www.usenix.org/conference/vehiclesec26/presentation/low

This paper is included in the Proceedings of the
4th USENIX Symposium on Vehicle Security and Privacy.

August 10–11, 2026 • Baltimore, MD, USA

ISBN 978-1-939133-56-4

Open access to the Proceedings of the
4th USENIX Symposium on Vehicle Security and Privacy
is sponsored by USENIX.

Charge It to My Neighbor: A Relay Attack on ISO 15118 Plug and Charge Payment

Jakob Löw
Technische Hochschule Ingolstadt

Vishwa Vasu
Technische Hochschule Ingolstadt

Thomas Hutzelmann
Technische Hochschule Ingolstadt

Hans-Joachim Hof
Technische Hochschule Ingolstadt

Abstract

ISO 15118, the leading standard for DC fast charging in Europe, includes a plug-and-charge mechanism that allows electric vehicles to handle payment automatically via contract certificates. We present a novel relay attack against this mechanism: an attacker builds a fake charging station, plugs it into a victim's vehicle, and relays the cryptographic authentication to a real charging station — charging the attacker's vehicle while billing the victim. The attack exploits the absence of station-identifying information in the plug-and-charge signature, combined with weaknesses in how ISO 15118 handles TLS certificates. We provide a proof-of-concept implementation demonstrating the full attack chain and discuss possible mitigations and alternatives. As plug-and-charge adoption grows, addressing this vulnerability is critical before it becomes widely exploitable.

1 Introduction

Electric vehicles are gaining in popularity. Especially in countries with corresponding incentives to buy electric over combustion engines, like China and Norway, they reach nearly 100% market share in newly registered vehicles [1]. With the increasing number of electric vehicles on the road also comes a growing demand for fast charging stations, allowing electric vehicles to quickly recharge before continuing their trip.

In Europe, the leading standard for charging communication between an electric vehicle and a charging station is ISO 15118 [13–15]. This standard not only describes the required communication between the two parties but also includes additional features, such as plug and charge payment, allowing the user to handle payment directly after plugging in a vehicle without further user interaction. A recent study found that plug and charge is the most preferred yet least used payment method among electric vehicle drivers, while mobile app and RFID card based payment are the most used but least preferred [23]. This significant discrepancy suggests that plug and charge adoption will continue to grow. Handling payment directly within the charging communication, however, also increases the attack surface and, especially, the reward potential that attackers could have when attacking charging communication.

In this paper, we present a novel relay attack against the plug and charge payment mechanism of ISO 15118. An attacker builds a fake charging station, plugs it into a victim's vehicle, and relays the cryptographic payment authentication to a real charging station — charging the attacker's vehicle while billing the victim. The attack exploits the absence of station-identifying information in the plug and charge signature, combined with weaknesses in how ISO 15118 handles TLS certificates. We provide a proof-of-concept implementation demonstrating the full attack chain and discuss possible mitigations, including an alternative out-of-band payment approach that avoids the vulnerability entirely.

While, to the best of our knowledge, the relay attack itself has not been described before, our proof-of-concept implementation builds on shortcomings in charging communication security identified in related work [11, 20–22]. Conti et al. [6] have previously demonstrated an attack against plug and charge payment; however, their approach relies on opaque forwarding, which introduces several limitations discussed later.

Section 2 provides background on the plug and charge mechanism and TLS certificate handling in ISO 15118. Section 3 discusses related work on charging communication security. Section 4 introduces the adversary model used for evaluating the vulnerability. Section 5 presents the plug and charge relay vulnerability. Section 6 describes our proof-of-concept implementation. Finally, Section 7 discusses mitigations and an alternative payment approach.

2 Background

ISO 15118 [13–15] defines the communication protocol between electric vehicles and charging stations for DC fast charging in Europe. The standard uses powerline communication over the charging cable, with IPv6 for network addressing and optionally TLS for session encryption. Before the main

charging communication begins, several initialization steps take place: low-level signaling per IEC 61851 [12], powerline network formation via the Signal Level Attenuation Characterization (SLAC) protocol, IPv6 address assignment, and TCP connection parameter exchange using the Service Discovery Protocol (SDP). Previous work has identified vulnerabilities in many of these initialization steps [9, 11, 16, 20]. For this paper, the relevant aspects of ISO 15118 are the plug and charge payment mechanism and the use of TLS certificates, both described below.

2.1 Plug and Charge

After the SDP request and response described in the previous section, the two communication parties establish a TCP connection. When TLS is supported by both parties, a standard TLS handshake follows. Within this established connection, the main charging communication is performed. The vehicle sends a `ServiceDiscoveryReq` to which the charging station responds with a `ServiceDiscoveryRes`. This response includes a list of supported payment methods. As of today, mostly the `ExternalPayment` method is used [18, 20]. With this method, the user has to handle payment directly with the charging station, for example, using a credit card.

Alternatively, the charging station may offer `ContractCertificate` payment. If the vehicle selects this payment method in its `PaymentServiceSelectionReq`, it follows up with its certificate in a `PaymentDetailsReq`. The charging station responds with a `PaymentDetailsRes`, including a unique challenge consisting of random bytes. The vehicle has to sign the challenge bytes using the private key belonging to the given certificate, sending the signature in a `AuthorizationReq`. This way, the vehicle proves to the charging station that it has ownership of the private key belonging to the contract certificate presented in the previous step. Figure 1 illustrates this communication flow, when plug and charge payment is used.

According to the ISO 15118 standard [13], plug and charge shall only be used when the connection is encrypted using TLS. For establishing a TLS connection with ISO 15118-2 [13], only the server needs to authenticate itself. With the newer ISO 15118-20 [15], mutual authentication is required for establishing an encrypted connection. With ISO 15118, there are two separate public key infrastructures (PKIs). One provides certificates to charging stations and vehicles for establishing TLS communication. While ISO 15118-2 requires only the charging station to possess a certificate, the newer ISO 15118-20 demands that both sides authenticate using a valid certificate obtained from the first PKI. The second PKI is responsible for handing out contract certificates used for payment. On a technological level, contract certificates used for payment are identical to certificates used for session authentication. While a vehicle typically has only one fixed certificate for session authentication, this separation enables it to have multiple, possibly user-configurable contract certificates.

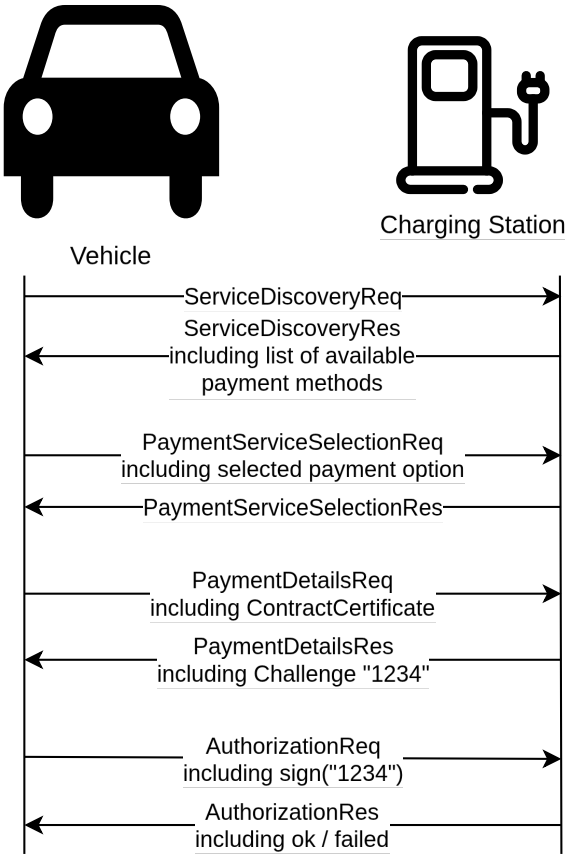


Figure 1: Plug and Charge Flow

2.2 TLS Certificate Handling in ISO 15118

For securing the main charging communication, the ISO 15118 standard uses Transport Layer Security. While TLS is a well-established protocol for securing and authenticating sessions, the way ISO 15118 employs it leaves gaps that are critical enablers for the relay attack presented in this paper.

TLS uses certificate-based authentication with a public key infrastructure for creating and signing certificates. On the web, where TLS is commonly used, this means there are a number of root certificate authorities. Website operators can acquire a certificate for their website from any of the commonly trusted certificate authorities. Web browsers usually bundle a large number of certificate authority root certificates and update them regularly. In the vehicle ecosystem, where storage space and internet connectivity are sparse, managing large certificate stores and especially updating them is significantly harder. As of today, there are multiple certificate authorities for charging communication; however, it is not clear to the public which OEMs ship their vehicles with trust for which authorities.

The most significant problem for the attack presented in this paper is the lack of verifiable charging station identification.

When on the web connecting to a website `example.com` using TLS, the web browser ensures the presented certificate was issued to `example.com`. For charging communication, the ISO 15118 standard requires the charging station ID to be encoded as the certificate identity. While this identity is unique per charging station, there is no way for the vehicle to check if the provided certificate belongs to the charging station currently connected to [20]. A potential attacker could acquire one certificate from any charging station and use this certificate to impersonate any other charging station. While certificate compromise can happen on the web as well, the impact is different: on the web, when a certificate for `example.com` is known to an attacker, the attacker can only impersonate this one website. With ISO 15118, when a certificate for one charging station is compromised, an attacker can impersonate any charging station, including stations from other vendors, operators or in other countries. This creates a weakest-link problem: the security of the entire ecosystem is bounded by the least secure charging station. Real-world evidence suggests that obtaining certificates from charging stations is feasible. Even Alpitronic, the largest DC fast charging hardware vendor in Europe, has been affected by vulnerabilities that could enable certificate extraction. In 2024, it was discovered that the configuration panel of Alpitronic Hyperchargers is accessible from the powerline communication network, meaning any connected vehicle can reach the charging station's administrative interface [24]. Additionally, a CISA advisory revealed that Alpitronic Hyperchargers shipped with default credentials for this administrative interface [5] (CVE-2024-4622). While neither vulnerability has been publicly demonstrated to lead to certificate extraction, they illustrate the kind of access an attacker could leverage, and the fact that they affect the market-leading vendor underscores the weakest-link concern.

3 Related Work

With the increasing adoption of electric vehicles, more and more cybersecurity researchers have uncovered vulnerabilities in ISO 15118. Most related work has focused on the initialization phases of charging communication, such as vulnerabilities in powerline communication [8, 11, 16], the SLAC process [9, 11], and service discovery [3, 19]. While these lower-layer vulnerabilities can facilitate man-in-the-middle positions that complement our attack (see Section 6.4), the relay vulnerability we present operates independently at the application layer.

The most closely related work is EVExchange by Conti et al. [6], which presents a relay attack on ISO 15118 charging communication. In EVExchange, an attacker places two malicious devices at two adjacent charging stations managed by the same backend infrastructure. These devices forward the entire network traffic between the two EVSEs, effectively swapping which vehicle communicates with which charging station. Since the relay operates at the network layer, it does not need to decrypt TLS-encrypted traffic; instead, it opaquely forwards all packets. The attacker can then control when to stop each vehicle's charge, making the victim pay for the attacker's energy consumption. EVExchange requires both the attacker and victim vehicles to be physically present at the same charging location, as the malicious devices must be installed between the vehicles and adjacent EVSEs. As a countermeasure, Conti et al. propose a distance bounding protocol that measures round-trip times of rapid packet exchanges to detect the additional latency introduced by the relay.

Because EVExchange relays the entire charging session at the network layer, the charging parameters sent by each vehicle must be consistent with the physical charging station it is actually connected to. In particular, during DC charging, the vehicle continuously reports its present battery voltage in `CurrentDemandReq` messages. Both vehicles must therefore have the same battery voltage for the relayed session to proceed without errors. Even when two vehicles start at the same voltage, the attacker's vehicle will see its battery voltage increase as it charges, causing a growing mismatch with the victim's reported values. The only way to avoid this inconsistency would be to charge both vehicles simultaneously, which eliminates the attacker's economic gain.

Table 1 summarizes the key differences between EVExchange and the vulnerability presented in this paper. In contrast to EVExchange, the plug and charge relay vulnerability presented in this paper (Section 5) operates at the application layer, specifically targeting the plug and charge authentication mechanism. Rather than forwarding entire network flows, our attack selectively relays the cryptographic messages used for payment authentication: the contract certificate, the challenge, and the signed response. This allows the attacker vehicle and the fake charging station to be at entirely different physical locations, communicating over a cellular network, similar to keyless go relay attacks on vehicles. Furthermore, while EVExchange requires tampering with existing charging infrastructure, our attack only requires the attacker to build a portable fake charging station that is plugged into the victim's vehicle.

4 Adversary Model

Bao et al. [3] have published an extensive threat analysis regarding ISO 15118, including the modeling of various adversaries. We utilize their adversary models for evaluating the impact of the described vulnerabilities and their potential exploitation by attackers. Of the adversaries defined by Bao et al., two are particularly relevant to the plug and charge relay vulnerability presented in this paper. Additionally, we introduce a third adversary, the *equipment supplier*, who is not defined by Bao et al. but is relevant for assessing the real-world scalability of the attack.

Table 1: Comparison of EVExchange and this work

Property	EVExchange [6]	This paper
Relay layer	Network (IP)	Application (ISO 15118)
Relayed data	All network traffic	Payment authentication messages
Vehicle Requirements	Same battery voltage	None
Actual Charging	Both vehicles	Only attacker vehicle
TLS broken	No (opaque forwarding)	No (separate sessions)
Countermeasure	Distance bounding	Context binding / backend

The *freeloader* aims to charge their own vehicle for free by making someone else pay for the charged energy. The freeloader has basic technical knowledge of electric vehicle charging and physical access to publicly accessible vehicles and charging stations. However, the freeloader does not have the capability to develop custom hardware or software and instead relies on readily available tools or equipment obtained from third parties. Using the relay attack described in Section 5, a freeloader could build or acquire a fake charging station, plug it into a victim's vehicle, and relay the payment authentication to a real charging station where the freeloader's vehicle is charging.

The *contract-sharer* aims to share one charging contract with a large number of other users, possibly abusing flat charging fees. Similar to the freeloader, the contract-sharer has basic technical knowledge and physical access to charging infrastructure, but does not necessarily possess advanced hardware or software development skills. The contract-sharer's distinguishing characteristic is the organizational capability to coordinate multiple users and distribute relay equipment or access credentials among them. By relaying the plug and charge authentication from a single contracted vehicle, a contract-sharer could enable multiple unauthorized vehicles to charge under the same contract.

The *equipment supplier* manufactures and sells ready-to-use relay equipment to customers such as freeloaders and contract-sharers. Unlike the other two adversaries, the equipment supplier possesses advanced technical knowledge of charging communication protocols, powerline communication, and embedded systems development. The equipment supplier is capable of developing custom hardware and software, such as portable fake charging stations and man-in-the-middle devices that combine known powerline communication vulnerabilities [9, 11, 16] with the plug and charge relay vulnerability described in Section 5. The equipment supplier's motivation is financial profit from selling attack equipment rather than directly exploiting the vulnerability. Similar to the market for keyless go relay devices used in vehicle theft, the existence of an equipment supplier would significantly lower the barrier to entry for freeloaders and contract-sharers, as they would no longer need any technical expertise to carry out the attack.

5 Plug and Charge Relay Vulnerability

As described in Section 2.1, ISO 15118 supports contract-based payment, also known as plug and charge. With this method, the vehicle transmits its contract certificate to the charging station, which responds with a random challenge. The vehicle then signs this challenge using the private key belonging to the contract certificate, proving ownership. Assuming the charging station uses a good random number generator and generates a new challenge for each session, this technique effectively prevents replay attacks. However, the shortcoming with the current mechanism defined by ISO 15118 is the absence of any identifying credential of the charging station for the vehicle to check. The vehicle does not include any additional information in the signed data, such as the current timestamp or information about the charging station it is connected to. Thus, while the challenge prevents replay attacks, it does not effectively prevent relay attacks. While Szakály et al. [20] already discussed the underlying TLS identification shortcomings that enable this attack, to the best of our knowledge, the plug and charge relay attack itself has not been previously described or implemented.

Normally, the use of TLS would prevent an attacker from creating a custom charging station emulator, as the lack of a valid certificate would prevent the victim vehicle from accepting it. However, due to the certificate shortcomings described in Section 2.2, an attacker could use certificates and private keys obtained from any compromised charging station to make the emulator impersonate any real charging station. Since a certificate from any vendor is accepted by all vehicles, compromising a single poorly secured charging station is sufficient to attack vehicles at any other station, regardless of that station's own security posture. This creates a weakest-link problem: the security of the entire ecosystem is bounded by the least secure charging station vendor. With ISO 15118-20, which mandates mutual TLS authentication, the attacker vehicle must additionally present a valid vehicle TLS certificate when connecting to the *regular charging station*. This means the attacker must also have compromised at least one vehicle's TLS certificate and private key. While this raises the bar compared to ISO 15118-2, payment handling is identical across both standard versions, so the fundamental relay vulnerability remains: the plug and charge signature still contains no

station-identifying information, and the attack proceeds in the same way once valid certificates for both sides are available.

To perform such a relay attack, an attacker could build a charging station emulator and plug it into a victim's vehicle. Simultaneously, the attacker vehicle is plugged into a regular charging station. After initializing the charging communication with the victim's vehicle, the vehicle sends its charging contract certificate. The attacker forwards this certificate to the charging station that the attacker's vehicle is plugged into, acting as if this certificate belongs to the attacker's vehicle. The charging station then generates a random challenge and sends it to the attacker's vehicle. Normally, the attacker would need the private key belonging to the victim's contract certificate in order to sign the challenge. However, the attacker simply forwards the challenge to the fake charging station, which sends it to the victim vehicle. The victim signs the challenge and returns the signature, which the attacker relays back to the regular charging station. Since the signed data contains no station-specific information, there is no way for the regular charging station to detect that this signature was originally created by a different vehicle for a different charging station. Figure 2 visualizes the communication flow of this relay attack.

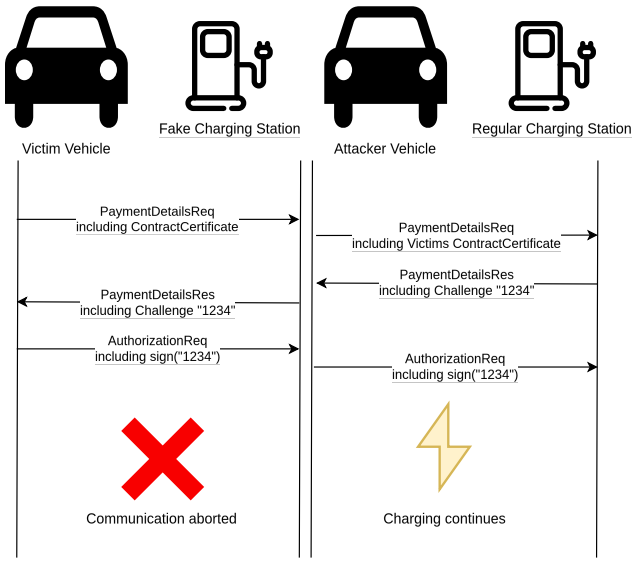


Figure 2: Plug and Charge Relay Attack

6 Implementing and Combining Vulnerabilities for a Payment Fraud Proof of Concept

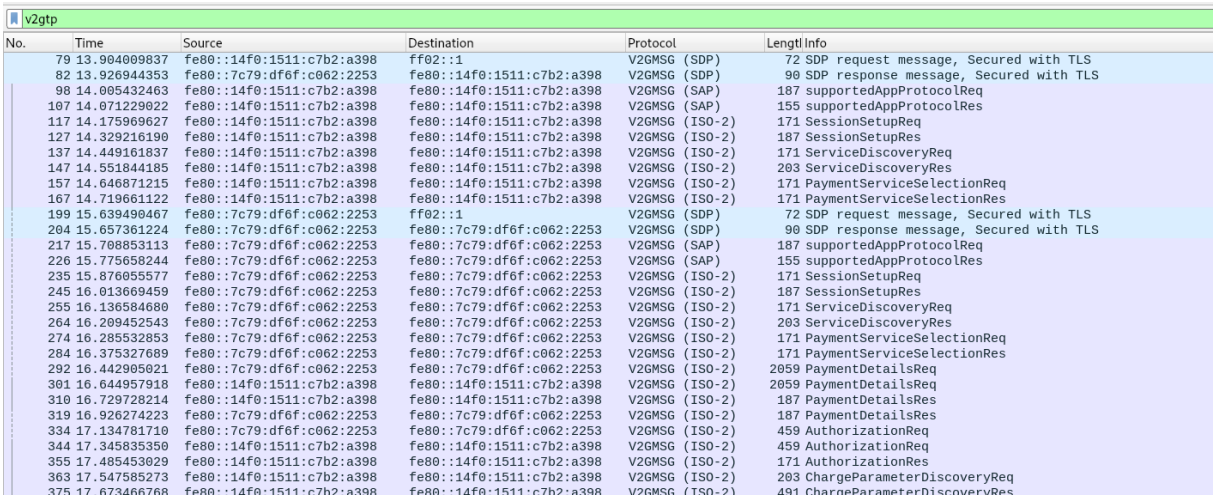
The previous section described a plug and charge relay vulnerability allowing an attacker to charge their vehicle while making a second vehicle, the victim vehicle, pay for the charging session. By combining this vulnerability with the vulnerability described in section 2.2, this section describes a possible proof of concept proving the feasibility of the attack. The software stacks, published as open source alongside this paper, can run entirely in a virtual environment, communicating with each other using local network interfaces. Additionally, the proof of concept was successfully tested using real hardware: our own charging station as the *regular charging station*, our EV simulator as the *victim vehicle*, and two powerline modems acting as *attacker vehicle* and *fake charging station*.

6.1 Communication Participants and Requirements

For this proof of concept, a custom PKI, as well as a set of virtual vehicles and virtual charging stations, is created. The following lists the created communication participants, which are also present in figure 2:

- The *victim vehicle*, which in reality could be any vehicle found parked on the street, after forcefully opening its charge port.
- The *attacker vehicle*. The goal of the attack is to charge the *attacker vehicle* with energy being billed to the *victim vehicle*.
- The *regular charging station*. This is the charging station that will be tricked into charging the *attacker vehicle* while billing the *victim vehicle* for the provided energy.
- The *compromised charging station* does not actually participate in the communication during the attack, but the attacker was previously able to extract this charging station's certificate and private key.
- The *fake charging station* is a purpose-built device. It does not need to actually provide power but merely needs to communicate with the *victim vehicle* to perform the plug and charge handshake.

In order to create the *attacker vehicle* as well as the *fake charging station*, a customized ISO 15118 stack implementation is required. Normally, when sending a `PaymentDetailsReq`, the vehicle includes its own certificate from its certificate store. For the attack to be successful, the *attacker vehicle* instead needs to send the certificate of the *victim vehicle*, which was received by the *fake charging station*. Similarly, when the *fake charging station* sends the `PaymentDetailsRes` to the *victim vehicle* the included challenge needs to be identical to the original challenge received by the *attacker vehicle* from the *regular charging station*. Additionally, the *attacker vehicle* will need to include the signature received by the *fake charging station* from the *victim vehicle* when sending its `AuthorizationReq` to the *regular charging station*. Therefore, a real-time communication between the *fake charging station* and the *attacker vehicle* is required, allowing them to exchange the certificate, challenge, and signature.



v2gtp

No.	Time	Source	Destination	Protocol	Lengtl	Info
79	13.904009837	fe80::14f0:1511:c7b2:a398	ff02::1	V2GMSG (SDP)	72	SDP request message, Secured with TLS
82	13.926944353	fe80::7c79:df6f:c062:2253	fe80::14f0:1511:c7b2:a398	V2GMSG (SDP)	90	SDP response message, Secured with TLS
98	14.005432463	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (SAP)	187	supportedAppProtocolReq
107	14.071229022	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (SAP)	155	supportedAppProtocolRes
117	14.175969627	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	171	SessionSetupReq
127	14.329216190	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	187	SessionSetupRes
137	14.449161837	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	171	ServiceDiscoveryReq
147	14.551844185	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	203	ServiceDiscoveryRes
157	14.646871215	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	171	PaymentServiceSelectionReq
167	14.719661122	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	171	PaymentServiceSelectionRes
199	15.639490467	fe80::7c79:df6f:c062:2253	ff02::1	V2GMSG (SDP)	72	SDP request message, Secured with TLS
204	15.657361224	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (SDP)	90	SDP response message, Secured with TLS
217	15.708853113	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (SAP)	187	supportedAppProtocolReq
226	15.775658244	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (SAP)	155	supportedAppProtocolRes
235	15.876055577	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	171	SessionSetupReq
245	16.013669459	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	187	SessionSetupRes
255	16.136584680	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	171	ServiceDiscoveryReq
264	16.209452543	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	203	ServiceDiscoveryRes
274	16.285532853	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	171	PaymentServiceSelectionReq
284	16.375327689	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	171	PaymentServiceSelectionRes
292	16.442905021	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	2059	PaymentDetailsReq
301	16.644957918	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	2059	PaymentDetailsReq
310	16.729728214	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	187	PaymentDetailsRes
319	16.926274223	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	187	PaymentDetailsRes
334	17.134781710	fe80::7c79:df6f:c062:2253	fe80::7c79:df6f:c062:2253	V2GMSG (ISO-2)	459	AuthorizationReq
344	17.345835350	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	459	AuthorizationReq
355	17.485453029	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	171	AuthorizationRes
363	17.547585273	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	203	ChargeParameterDiscoveryReq
375	17.673466768	fe80::14f0:1511:c7b2:a398	fe80::14f0:1511:c7b2:a398	V2GMSG (ISO-2)	491	ChargeParameterDiscoveryRes

Figure 3: Wireshark Log of the two charging sessions during the attack

6.2 Proof of Concept Implementation Details

In reality, this communication is likely to be handled via a cellular network, allowing the *fake charging station* to be at a different location than the *attacker vehicle*. In the proof of concept, the *attacker vehicle* and *fake charging station* exchange the certificate, challenge, and signature via a TCP connection, but run on the same machine. If the *attacker vehicle* reaches the step where it should send a `PaymentDetailsReq` before the certificate is available from the *fake charging station*, it simply stalls and waits for it to arrive over the network. For `PaymentDetailsReq` and `AuthorizationReq`, ISO 15118 specifies a response timeout of 2 s [13]. Since the relay only needs to forward small application-layer messages, a cellular network with typical round-trip times well below 100 ms would provide more than enough margin for the relay to complete within the protocol's timeout constraints.

Implementation-wise, the proof of concept is based on the ISO 15118 stack formerly developed and published by SwitchEV, now EcoG [10]. EcoG is a company selling commercial communication controllers and software for charging stations. While the software stack is published as open source, allowing for easy modifications and attack implementation, it is also deployed in real charging stations. In addition to the required changes for handling `PaymentDetailsReq`, `PaymentDetailsRes` and `AuthorizationReq` packets, the customized software stack also creates SSL key logs, allowing for easy inspection of encrypted traffic in Wireshark [25]. Our modified software stack is published under the original software license, allowing others to easily reproduce the attacks or try them against their own ISO 15118 implementations: https://github.com/securityinmobility/plug-and-charge-relay-poc

In order to run the proof of concept, in addition to the customized software stacks used for the *fake charging station* and *attacker vehicle*, two more ISO 15118 instances are required for the *regular charging station* and *victim vehicle*. In a purely virtual setup, the same ISO 15118 stack can be used in its original unmodified version. For our hardware-based test, a real charging station serves as the *regular charging station* and our EV simulator acts as the *victim vehicle*, while two powerline modems running the customized stack act as *attacker vehicle* and *fake charging station*. Since the relevant vulnerabilities described in section 2.2 and section 5 are not specific to one implementation, but rather apply to all implementations of the standard, in theory, any compliant ISO 15118 implementation could be used.

6.3 Proof of Concept Evaluation

Running the attack proof-of-concept produces a traffic log similar to the one shown in figure 3. In this screenshot, the IPv6 address of the *attacker vehicle* ends on `a398`, while the IPv6 address of the *victim vehicle* ends on `2253`. At first, the *attacker vehicle* initiates a charging session with the *regular charging station*, but stalls after the `PaymentServiceSelectionRes`. Normally, the *attacker vehicle* would follow up with a `PaymentDetailsReq`, but it has to wait for the content from the *victim vehicle* to relay the message. The charging session between *victim vehicle* and *fake charging station* is started slightly later. As soon as the *fake charging station* receives the `PaymentDetailsReq` the included certificate is also used by the *attacker vehicle* for its packet. Similarly, as soon as the *attacker vehicle* receives the `PaymentDetailsRes` including the challenge required to sign, the *fake charging station* sends exactly this challenge to the *victim vehicle*. After the signed challenge is received from the *victim vehicle* the communication with the victim stops. Only the charging session with the *attacker vehicle* continues.

6.4 From Proof of Concept to the Real World

As described in Section 4, the *freeloader* and the *contract-sharer* are the two adversaries which could directly benefit from abusing the described plug and charge relay vulnerability.

We successfully executed the attack both in a purely virtual setup and on real powerline hardware, as described in Section 6. In this subsection, we discuss the additional engineering that would be required to extend the attack from our controlled hardware setup to unmodified, off-the-shelf production vehicles encountered in the wild. While our hardware-based test already demonstrates the attack using real charging infrastructure, a real-world scenario requires targeting actual vehicles rather than an EV simulator. Building the *fake charging station* which performs high-level communication with the *victim vehicle* can easily be done using openly available charging communication evaluation kits [4, 17]. Modifying the charging communication stack inside a vehicle, to build the *attacker vehicle*, however, is a more complicated task. A simpler approach would be to utilize known powerline communication vulnerabilities to design a man-in-the-middle device, which modifies the communication between an unmodified *attacker vehicle* and a *regular charging station*. Previous work has demonstrated wireless powerline interception [16], network infiltration via SLAC sniffing [11], and man-in-the-middle attacks on charging communication [9, 22]. Combining these techniques allows designing a wireless charging communication man-in-the-middle device. This device could then perform plug and charge authentication with the charging station on one side and behave like a free charging station towards the *attacker vehicle*. Similar to keyless go relaying hardware utilized by car thieves, these known lower-layer vulnerabilities could be combined with the relay vulnerability presented in this paper to develop a pair of devices able to perform plug and charge relay attacks. As described in Section 4, this development and packaging of attack equipment into a consumer-ready product is the role of the *equipment supplier*, whose existence would make the attack accessible to freeloaders and contract-sharers without any technical expertise.

7 Possible Mitigations

This section discusses approaches for mitigating the plug and charge relay vulnerability presented in Section 5, as well as an alternative payment mechanism that avoids the vulnerability entirely.

7.1 Improving TLS Certificates for Charging Communication

For the plug and charge relay to work, the attacker needs to be able to impersonate a valid charging station towards the victim vehicle. Due to the TLS weakness described in Section 2.2, a single compromised charging station certificate is sufficient to impersonate any station worldwide. Fixing this does not directly address the relay vulnerability itself, but it would prevent this attack and potential future vulnerabilities from being exploited at scale, as the ability to impersonate arbitrary charging stations is a prerequisite for most attack scenarios. Szakály et al. [20] already proposed two possible remedies for the TLS vulnerability: stricter OCSP checking and encoding geo-coordinates into charging station TLS certificates.

OCSP provides a mechanism to revoke compromised certificates, which would limit the window during which an attacker can abuse a stolen charging station certificate. However, OCSP checking is not mandatory for TLS leaf certificates in the plug and charge specification, and vehicles are not required to perform it. Making OCSP verification mandatory for vehicles would ensure that revoked charging station certificates can no longer be used for impersonation, provided that compromises are detected and reported in a timely manner.

Another way to strengthen the TLS trust model would be to encode geo-coordinates into charging station TLS certificates. Vehicles can then check the distance between their last GNSS position and the coordinates in the certificate. When an attacker obtains a compromised certificate, it would only allow impersonation of stations near the compromised station's location, rather than any station worldwide. This mitigation could be carried out as a TLS certificate extension, allowing backwards compatibility with vehicles that do not yet support the verification of geo-coordinates.

7.2 Timing-Based Detection

Distance bounding protocols that detect relay attacks through round-trip time measurements are well established in domains such as NFC and passive keyless entry. Conti et al. propose a similar approach as a countermeasure to EVExchange [6], measuring the latency of rapid challenge-response exchanges over the charging cable to detect the additional delay introduced by a relay. In principle, such timing analysis could also be applied to detect the plug and charge relay presented in this paper. However, production EV and EVSE controllers vary widely in processing power and system load, and the PLC channel over the charging cable is subject to variable latency from electrical noise and cable quality. This makes it difficult to establish reliable timing thresholds that distinguish relay-induced delay from normal variation across the diverse ecosystem of charging hardware.

7.3 Context-Binding the Plug and Charge Signature

The relay attack succeeds because the plug and charge signature covers only the random challenge issued by the charging station. Since the signed data contains no information about

which charging station the vehicle believes it is connected to, the signature is valid at any charging station, enabling the relay.

A direct countermeasure is to bind the signature to the context of the current session by including the charging station's EVSE ID in the signed data. During the TLS handshake, the charging station presents its certificate, which contains the EVSE ID as the certificate subject. The vehicle can extract this identifier and concatenate it with the challenge bytes before computing the signature. On the charging station side, the station concatenates its own EVSE ID with the challenge it issued and verifies the signature against the result.

This defeats the relay attack as follows: When the *victim vehicle* signs the challenge, it includes the EVSE ID of the *fake charging station* as presented during the TLS handshake. When the *attacker vehicle* forwards this signature to the *regular charging station*, the station verifies the signature against its own EVSE ID concatenated with the challenge. Since the two EVSE IDs differ, the signature verification fails and the authentication is rejected. The attacker cannot correct this mismatch without access to the victim's private contract certificate key.

This mitigation can be implemented in a backwards-compatible manner without changing the message format defined in ISO 15118. The challenge issued by the charging station is 16 bytes long. Instead of filling these bytes entirely with random data, an updated charging station embeds its country code and operator identifier (5 bytes) within the challenge, using the remaining 11 bytes for random data. Updated vehicles extract the EVSE ID from the charging station's TLS certificate and verify that the corresponding country and operator bytes in the challenge match; if they do not, the vehicle rejects the session, defeating the relay attack.

This approach is backwards compatible in both directions: vehicles that have not been updated treat the challenge as opaque random data and sign it as before, so they continue to function with updated charging stations. Conversely, updated vehicles connecting to a legacy charging station that issues fully random challenges can detect the absence of the expected EVSE identifier and reject the session. As a result, updated vehicles are protected regardless of whether the charging station has been updated, while legacy vehicles retain full functionality with both legacy and updated infrastructure.

7.4 Replacing Plug and Charge with Out-of-Band Payment

An alternative to fixing the plug and charge mechanism is to bypass it entirely. After establishing a TLS connection with the charging station, the vehicle forwards the EVSE ID to a manufacturer-provided backend, which initiates the charging session via the OCPP `RemoteStartTransaction` command — the same mechanism already used by mobile charging apps today. This eliminates the need for the cross-vendor PKI required by plug and charge, and requires no changes to ISO 15118, OCPP, or the charging station software. Additionally, this approach would greatly reduce complexity, as it removes the need for enrolling contract certificates and designing custom challenge-response procedures for plug and charge.

This approach has two drawbacks: it makes the vehicle manufacturer a gatekeeper for payment, and it requires an internet connection, unlike plug and charge which can operate offline. However, ISO 15118 already gives manufacturers control over which contract certificates are enrolled, and the trend towards connected charging infrastructure — reinforced by regulations such as the EU Alternative Fuels Infrastructure Regulation [2] — makes the offline limitation increasingly marginal. A full evaluation of this alternative payment approach, including its security properties, deployment considerations, and impact on existing roaming agreements, is beyond the scope of this paper and should be explored in more comprehensive studies in future work.

8 Conclusion

In this paper, we presented a novel relay attack against the plug and charge payment mechanism of ISO 15118. The attack exploits the absence of station-identifying information in the plug and charge signature, combined with the inability of vehicles to verify charging station identity through TLS certificates. As shown in our proof-of-concept implementation, these weaknesses make it possible to build a relay device, similar to keyless go relaying, that allows adversaries to charge a vehicle while a different vehicle is billed for the charged energy.

We discussed direct mitigations for the relay vulnerability, including binding the plug and charge signature to the charging station identifier and timing-based detection of relay delays. Additionally, we proposed an alternative out-of-band payment approach that shifts payment handling to the vehicle manufacturer's backend, reusing existing remote start mechanisms. This approach removes the need for the complex cross-vendor PKI required by plug and charge and can be implemented without changes to ISO 15118 or charging station software.

As plug and charge adoption grows, addressing this vulnerability is critical before it becomes widely exploitable.

9 Ethical Considerations

We disclosed the plug and charge relay vulnerability to CharIN, the organization managing the Combined Charging System and the ISO 15118 standard, prior to publication. In addition, we contacted several major charging station manufacturers directly; while they acknowledged the disclosure, they did not consider immediate mitigation necessary on their

side. As plug and charge is not widely deployed today [18,20], the immediate risk to users is limited. Furthermore, performing the attack in practice requires obtaining a valid TLS certificate trusted by the victim vehicle, which requires exploiting and extracting one from a real-world charging station, potentially using a zero-day exploit.

10 Open Science

Our proof-of-concept implementation is publicly available at: https://github.com/securityinmobility/plug-and-charge-relay-poc Apart from the modified charging communication stack, this repository also includes network dumps of successful demonstrations of the plug and charge payment fraud attack. By using the free and open source dissector for charging communication "dsV2Gshark" [7], these files can be viewed using Wireshark [25].

Acknowledgment

This work was created in the research project "Elektromobiles Sicheres Laden" (ESiLa) funded by the Bavarian Ministry of Economic Affairs, Regional Development and Energy under grant DIK0512/01.

References

[1] Open EV Charts. https://open-ev-charts.org/. Accessed: 30 June 2025.

[2] Regulation (EU) 2023/1804 of the European Parliament and of the Council of 13 September 2023 on the deployment of alternative fuels infrastructure, and repealing Directive 2014/94/EU (Text with EEA relevance), September 2023. Legislative Body: CONSIL, EP.

[3] Kaibin Bao, Hristo Valev, Manuela Wagner, and Hartmut Schmeck. A threat analysis of the vehicle-to-grid charging protocol ISO 15118. *Computer Science - Research and Development*, 33(1):3–12, February 2018.

[4] Chargebyte GmbH. Iso 15118 and din 70121 compliant dc charging controller for electric vehicle charging stations (evse) and electric vehicles (ev). https://chargebyte.com/controllers-and-modules/evse-controllers/evacharge-se. Accessed: 2025-03-06.

[5] CISA. Alpitronic Hypercharger EV Charger, May 2024. ICSA-24-130-02, CVE-2024-4622.

[6] Mauro Conti, Denis Donadel, Radha Poovendran, and Federico Turrin. EVExchange: A Relay Attack on Electric Vehicle Charging System. In Vijayalakshmi Atluri, Roberto Di Pietro, Christian D. Jensen, and Weizhi Meng, editors, *Computer Security – ESORICS 2022*, pages 488–508, Cham, 2022. Springer International Publishing.

[7] dSpace Group. Free wireshark plugin to analyze iso15118 v2g communication between electric vehicles (ev) and charging stations (evse). https://github.com/dspace-group/dsV2Gshark. Accessed: 2025-03-06.

[8] Sébastien Dudek. HomePlugAV PLC: Practical attacks and backdooring. *Netw. Anal.*, 2015.

[9] Sébastien Dudek, Jean-Christophe Delaunay, and Vincent Fargues. V2g injector: Whispering to cars and charging units through the power-line. In *Proceedings of the SSTIC (Symposium sur la sécurité des technologies de l'information et des communications)*, pages 1–26, Rennes, France, 2019. SSTIC.

[10] EcoG GmbH. Implementation of the iso 15118 communication protocol (-2, -20, -8). https://github.com/EcoG-io/iso15118. Accessed: 2025-03-06.

[11] Lukas Eder, Jakob Löw, and Hans-Joachim Hof. Charging Communication Sniffing and Man-in-the-Middle Attacks. In *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems*, E-Energy '25, pages 799–804, New York, NY, USA, June 2025. Association for Computing Machinery.

[12] IEC. IEC 61851-1 ed2.0: Electric vehicle conductive charging system - Part 1: General requirements, 2010.

[13] ISO/IEC. ISO/IEC DIS 15118-2: Road vehicles - Vehicle to grid communication interface – Part 2: Network and application protocol requirements, 2012.

[14] ISO/IEC. ISO/IEC DIS 15118-3: Road vehicles - Vehicle to grid communication interface – Part 3: Physical and data link layer requirements, 2012.

[15] ISO/IEC. ISO/IEC DIS 15118-20: Road vehicles - Vehicle to grid communication interface – part 20: 2nd generation network layer and application layer requirements, 2022.

[16] Sebastian Köhler, Richard Baker, Martin Strohmeier, and Ivan Martinovic. Brokenwire: Wireless Disruption of CCS Electric Vehicle Charging. In *Proceedings 2023 Network and Distributed System Security Symposium*, San Diego, CA, USA, 2023. Internet Society.

[17] Jakob Löw, Dominik Bayerl, Kevin Mayer, and Hans-Joachim Hof. DrainDead: Emptying Batteries of Parked Electric Vehicles. In *3rd USENIX Symposium on Vehicle Security and Privacy*, pages 233–241, 2025.

[18] Jakob Löw, Kevin Mayer, and Hans-Joachim Hof. *Fast Charging Communication and Cybersecurity: A Technology Review*. November 2024.

[19] Marc Mültin. ISO 15118 as the Enabler of Vehicle-to-Grid Applications. In *2018 International Conference of Electrical and Electronic Technologies for Automotive*, pages 1–6, July 2018.

[20] Marcell Szakály, Sebastian Köhler, and Ivan Martinovic. Current Affairs: A Security Measurement Study of CCS EV Charging Deployments, February 2025. arXiv:2404.06635 [cs].

[21] Marcell Szakály, Sebastian Köhler, and Ivan Martinovic. Short: PIBuster: Exploiting a Common Misconfiguration in CCS EV Chargers. pages 243–249, 2025.

[22] Marcell Szakály, Martin Strohmeier, Ivan Martinovic, and Sebastian Köhler. DCeption: Real-world Wireless Man-in-the-Middle Attacks Against CCS EV Charging, January 2026. arXiv:2601.15515 [cs].

[23] USCALE. eMobility Charging Payment Study 2026, 2025.

[24] Wilco van Beijnum and Sebastiaan Laro-Tol. Hacking EV charging stations via the charging cable, 2024. Hardwear.io Netherlands 2024, The Hague.

[25] Wireshark Foundation. The world's most popular network protocol analyzer. https://www.wireshark.org/. Accessed: 2025-03-06.