

Fingerprinting ISO 15118 Charging Communication Stacks: A Two-Sided Measurement of Charging Stations and Vehicles

Jakob Löw

Technische Hochschule Ingolstadt
Ingolstadt, Germany
email: jakob.loew@thi.de

Hans-Joachim Hof

Technische Hochschule Ingolstadt
Ingolstadt, Germany
email: hof@thi.de

Abstract—ISO 15118 secures electric-vehicle charging communication with Transport Layer Security (TLS), which is also the trust anchor for Plug & Charge payment. Deployment is measured only as a bit: prior work reports *whether* a station offers TLS, never *what* it runs. We present the first TLS stack fingerprinting study of ISO 15118, on both sides of the connector. From 45 power-line traces recorded at 41 public charging stations in Germany, and 82 traces recorded against 11 production vehicles, we extract a layered passive fingerprint spanning the link layer, the discovery protocol, the handshake and the certificate chain, and turn it into implementation attributions using a reference library built by replaying a field-recorded ClientHello against 14 configurations of six TLS implementations, probing no live infrastructure. Ten of 41 stations (24.4 %) advertise TLS; three yield handshakes, producing three distinct fingerprints that separate three vendors sharing one certificate authority. One matches OpenSSL byte-for-byte, one is attributable by elimination, and one — built in 2024 — matches no tested implementation and returns an empty ServerHello extension block, negotiating neither renegotiation indication, nor the extended master secret, nor encrypt-then-MAC on a block-cipher suite. On the vehicle side, two fingerprints cover three models: both offer the cipher suite the standard mandates and no station selected, both reject an untrusted chain but one alerts and the other closes silently, and neither supports TLS 1.3. Two stations serve whichever security level the client requests, so an availability figure depends on what the measuring client asked for, and TLS 1.2 sessions disclose the charge point identifier in cleartext before authentication.

Keywords—electric vehicle charging; ISO 15118; transport layer security; protocol fingerprinting; measurement study.

I. INTRODUCTION

Direct-current (DC) fast charging in Europe uses the Combined Charging System (CCS). The high-level communication between the Electric Vehicle Communication Controller (EVCC) and the Supply Equipment Communication Controller (SECC) runs over HomePlug Green PHY Power-Line Communication (PLC) and is specified by ISO 15118-2 [1] and, increasingly, ISO 15118-20 [2], with DIN SPEC 70121 [3] still widely deployed as a plaintext fallback. ISO 15118-2 protects the session with TLS, in which the SECC is the TLS server and the vehicle an anonymous client; ISO 15118-20 upgrades this to mutually authenticated TLS 1.3. TLS is also the precondition for Plug & Charge, the contract-certificate-based payment flow whose relay and man-in-the-middle exposure has been repeatedly demonstrated [5][7][8][14].

Existing measurement studies answer a binary question. Szakály et al. found that 12% of 325 European stations

implement TLS at all [4]; a market-analysis-driven study put it at 27.4% across a scope covering some 52% of the 40,000-plus CCS charge points registered in Germany [6]. Neither tells us which implementation runs behind the socket or how it is configured — and a station commissioned in 2024 may ship a TLS configuration the web abandoned a decade earlier without any pass/fail conformance report revealing it.

Server-side TLS fingerprinting is well established on the web [10][11] but has not, to our knowledge, been applied to ISO 15118, where the setting is unusually favourable: stacks are few and embedded, the Public Key Infrastructure (PKI) is centralised, and the client is a vehicle rather than a browser, so the ClientHello that elicits a fingerprint can be held fixed by construction.

Our contributions are threefold:

- 1) A *layered passive fingerprint* for ISO 15118 endpoints, combining link-layer, discovery, handshake and certificate features. All of them are obtainable from one unmodified charging attempt.
- 2) A *reference fingerprint library* for six TLS stacks in 14 configurations, built by replaying a field-recorded ClientHello. It converts observed fingerprints into implementation attributions, including attribution by elimination, and quantifies their stability.
- 3) A *two-sided empirical characterisation* of TLS as deployed. It covers station availability, stack attribution, a client-selected security level and a cleartext identity leak, together with the vehicle-side fingerprints, rejection behaviour and version ceiling that determine what a station can actually negotiate in the field.

The scope is deliberately narrow: we measure what two existing corpora contain, and attribute implementations only up to indistinguishability from a tested stack. Section V states these constraints in full.

The rest of the paper is structured as follows. Section II covers ISO 15118 transport security, published attacks and TLS fingerprinting. Section III describes the corpora and the replay methodology. Section IV presents the results, Section V discusses implications and limitations, and Section VI concludes.

II. BACKGROUND AND RELATED WORK

This section covers the three bodies of work the paper builds on: the transport security ISO 15118 defines, the attacks published against it, and TLS fingerprinting as a technique.

ISO 15118 transport security. After link establishment by Signal Level Attenuation Characterisation (SLAC), the EVCC discovers the SECC with the SECC Discovery Protocol (SDP) over User Datagram Protocol (UDP) port 15118. The SDP request carries a one-byte *security* field (0x00: TLS, 0x10: no TLS); the response returns the SECC’s link-local IPv6 address, TCP port, and the security level it will serve. ISO 15118-2 mandates `TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256` (0xc025) with server-only authentication against the Vehicle-to-Grid (V2G) PKI, rooted commercially — in Europe predominantly at Hubject. ISO 15118-20 requires TLS 1.3 with `TLS_AES_256_GCM_SHA384` and mutual authentication.

Attacks. The adversary model is undemanding; Eder et al. [8] obtain a usable signal by twisting sniffer wires around the charging cable, and note that the HomePlug network membership key travels in the clear during SLAC, so link-layer encryption gives no confidentiality; Dception [5] does the same wirelessly with a Software-Defined Radio (SDR). Brokenwire disrupts charging remotely [13]; EVExchange [14] and a relay attack on Plug & Charge [7] exploit the vehicle’s inability to bind the authenticated session to the charger it is plugged into. All presuppose, or are amplified by, weak or absent TLS.

TLS fingerprinting. JA3 and JA3S hash the ordered TLS version, cipher and extension lists of the ClientHello and ServerHello, respectively [10]. JA4S, the server counterpart of the JA4+ suite, encodes the negotiated version, extension count and cipher readably alongside a hash of the extension list [11]. Client-side fingerprints are known to be coarse; the server side of an embedded, vendor-partitioned ecosystem is a far better fit.

What this paper adds. Prior ISO 15118 measurements treat TLS as a binary property of a station, and prior fingerprinting work targets web clients. We treat the handshake as a structured observation that identifies an implementation, and we measure both endpoints, so what a station offers can be compared against what vehicles request.

III. DATASET AND METHODOLOGY

Dataset. We analyse the public trace corpus accompanying [6]: 45 pcapng captures recorded between January and April 2026 at public DC charging sites in Bavaria, Germany, together with a ground-truth table giving Charge-Point Operator (CPO), hardware manufacturer, model and build year for each capture. Captures were taken with a custom EVCC attached to the CCS connector; every capture is a genuine, operator-sanctioned charging attempt. Two captures never reach the IP layer and are excluded; the remaining 43 contain 41 distinct SECC link-local identifiers, which we treat as 41 stations. The corpus is skewed towards Alpitronic HYC hardware (31 stations), consistent with its reported $\approx 66\%$ share of German CCS charge points [6]; ABB (4), Compleo (3), ads-tec (2) and Tritium (1) make up the remainder.

Vehicle-side corpus. We additionally analyse the 82 traces published in [9], recorded with an Electric Vehicle Supply Equipment (EVSE) emulator holding no valid V2G credential.

The 11 vehicles are those that the study could obtain access to — a convenience sample spanning seven manufacturers, not a designed selection. The vehicles are the TLS clients and every handshake fails, which is what makes the corpus useful: it captures each vehicle’s ClientHello and rejection behaviour with no station-side variable.

Feature extraction. Restricting every analysis to link-local IPv6, we extract, per endpoint, the SECC hardware address and its Organisationally Unique Identifier (OUI) from the link-local interface identifier; the SDP security byte offered and returned, and the announced SECC port; the negotiated version, cipher, and ServerHello or ClientHello extension list and order, from which we compute JA3, JA3S and JA4S; and the full X.509 chain with its extension inventory. TLS records are reassembled per TCP stream and parsed with an independent parser rather than a protocol dissector, whose JSON output collapses repeated keys and silently truncates cipher and extension lists — precisely the fields a fingerprint depends on. That parser and the extraction scripts around it were written with the assistance of Claude Cowork; they are released, together with the outputs behind every number reported here, at <https://github.com/securityinmobility/charging-tls-fingerprinting>.

Reference library by ClientHello replay. A JA3S value is meaningful only relative to the ClientHello that elicited it, so a server fingerprint cannot be looked up in a web-derived database. Here, that constraint is an advantage: the same EVCC produced every ClientHello in the station corpus, so an exactly comparable library can be built offline. We extract the 517-byte ClientHello record verbatim from the capture and send it unmodified over a socket to OpenSSL 3.0.13, GnuTLS 3.8.3, mbedTLS 2.28.8, wolfSSL 5.6, Go 1.24 `crypto/tls` and Java 21 JSSE in 14 configurations, each holding a P-256 Elliptic Curve Digital Signature Algorithm (ECDSA) certificate, repeating every measurement eight times to assess stability. The handshake is not completed — the recorded client’s keys are unavailable — but the server’s reply up to ServerHelloDone is all a fingerprint needs. The “no SNI” rows of Table II are the one exception, using a hello with the SNI extension removed and its lengths recomputed. No charging station is probed, making the method safe for safety-relevant infrastructure and repeatable by any reader with the published traces. The harness and the six reference servers were likewise built with the assistance of Claude Cowork.

IV. RESULTS

We report the results in the order an observer meets them during a charging attempt: discovery and availability, the server handshake and its attribution, the stability of those fingerprints, the certificate chain, and finally the vehicle side.

A. TLS availability and the SDP security byte

Ten of 41 stations (24.4%) answered at least one SDP request with `security = 0x00`, closely matching the 27.4% extrapolated in [6]. The other 31 answered `0x10` even when TLS was explicitly requested, advertising their lack of transport security honestly.

TABLE I. OBSERVED SECC FINGERPRINTS. ALL CHAINS ISSUE FROM THE SAME CA.

Hardware (build)	Ver./cipher	SH ext.	JA4S
F1 Tritium TRI105 (2019)	1.2 / 0xc023	65281-11-35-22-23	t120500_c023
F2 ABB HP CP500 C (2024)	1.2 / 0xc023	<i>none</i>	t120000_c023
F3 Alpitronic HYC-400 (2023)	1.3 / 0x1302	43-51	t130200_1302

Two of the ten — an Alpitronic HYC-300 and the Tritium unit — answered 0x00 to a TLS request and 0x10 to a plaintext request from the same EVCC within the same measurement campaign. These stations serve whichever level the client asks for. That is backwards compatibility behaving as specified rather than a defect: ISO 15118-2 carries the security byte precisely so that an EVCC can select its transport, and DIN SPEC 70121 offers no TLS at all, so a station that still accepts older vehicles has to honour a plaintext request. How an unauthenticated SDP exchange can be abused is a separate and already studied question [8][12].

What the observation does affect is measurement. Because the byte is client-selected, a single observed session cannot distinguish a station that will not do TLS from one that was never asked, and the two stations above would be recorded as plaintext-only by any passive observer who happened to watch a DIN session. Probing both values, as the corpus does, is therefore a prerequisite for an availability figure to mean anything — including the 24.4 % above.

The announced SECC TCP port is a reliable vendor discriminator before a single TLS byte is exchanged: 0xc7a6/0xc7a7 (51110/51111) for Alpitronic, 0xd121 (53537) for ABB, 0xc7ed (51181) for Compleo, 0xc000 (49152) for ads-tec. Tritium alone randomises its port per session (four observed), itself a distinguishing behaviour. Link-layer identification is weaker: 33 of 41 SECCs use locally administered MACs, leaving no OUI; the eight global addresses resolve to Microchip (five — the QCA700x PLC modem exposed as the SECC interface), congatec, ads-tec and one unassigned block.

B. Server fingerprints and stack attribution

Three stations in the corpus yielded a complete TLS handshake and form our fingerprinting sample. They produce three mutually distinct JA3S and JA4S values (Table I): the fingerprint separates all three hardware vendors, even though all three present certificates from the same CA. Table II gives the reference library.

F1 (Tritium) matches OpenSSL 3.0.13 restricted to 0xc023 exactly in both JA3S and JA4S, corroborated by two details the hash omits: encrypt-then-MAC, which OpenSSL negotiates only for CBC suites, and a NewSessionTicket. We attribute this 2019-built SECC to an OpenSSL-family stack.

F3 (Alpitronic HYC-400) matches two entries: OpenSSL default and Java 21 JSSE with the SNI removed. The field ClientHello, however, *contains* an SNI that JSSE rejects (Section IV-F); this station accepted it, so JSSE is excluded and F3 attributes to OpenSSL 3.x or a compatible stack. GnuTLS, wolfSSL, Go and mbedTLS are excluded by cipher or extension order. Attribution thus succeeds by elimination even where two

TABLE II. REFERENCE FINGERPRINT LIBRARY: SERVERHELLO ELICITED FROM EACH STACK BY THE FIELD-RECORDED CLIENTHELLO. “NO SNI” REPEATS THE MEASUREMENT WITH THE RFC-6066-ILLEGAL SNI STRIPPED.

Stack	Configuration	JA4S (truncated)
OpenSSL 3.0.13	default	t130200_1302_a56c
OpenSSL 3.0.13	max TLS 1.2	t120400_c02c_12a2
OpenSSL 3.0.13	TLS 1.2, 0xc023	t120500_c023_a944
OpenSSL 3.0.13	..., no tickets	t120400_c023_8cb4
GnuTLS 3.8.3	default	illegal_parameter
GnuTLS 3.8.3	no SNI	unstable (see text)
mbedTLS 2.28.8	default	t120300_cca9_5935
wolfSSL 5.6	default	t130200_1301_234e
Go 1.24	default	t130200_1301_a56c
Go 1.24	max TLS 1.2	t120400_c02b_9976
Java 21 JSSE	default	illegal_parameter
Java 21 JSSE	no SNI	t130200_1302_a56c

candidates hash identically — the value of a replay library over a lookup table.

F2 (ABB HP CP500 C, built 12/2024) matches nothing in the library and is anomalous in a security-relevant way: its ServerHello carries *no extensions at all*. The station therefore does not echo the RFC 5746 [19] renegotiation indication although the client offered TLS_EMPTY_RENEGOTIATION_INFO_SCSV, does not negotiate the extended master secret (RFC 7627) [20], and does not negotiate encrypt-then-MAC (RFC 7366) [21] although the selected suite 0xc023 is CBC-based — leaving the record layer in the MAC-then-encrypt configuration targeted by Lucky Thirteen-class attacks [15] and without the binding that mitigates triple-handshake attacks [16]. Every one of these mitigations predates the unit’s 2024 build date by at least nine years. We read F2 as a proprietary or reduced embedded stack — a diagnosis requiring one passive observation.

Neither TLS 1.2 station selected the ISO 15118-2 mandatory suite 0xc025; both chose the ephemeral-ECDH variant 0xc023, a deviation that *improves* security by adding forward secrecy — an illustration of how a conformance-only view misses security-relevant facts in both directions. Neither sent a CertificateRequest, consistent with ISO 15118-2’s unilateral authentication.

C. Fingerprint stability

Over eight repetitions with an identical ClientHello, OpenSSL, mbedTLS, wolfSSL, Go and JSSE each produced exactly one JA3S. GnuTLS produced two: it permutes its ServerHello extension order as an anti-ossification measure. A fingerprint database must therefore record a stack’s permutation behaviour alongside its hash, or a GnuTLS-based SECC will appear as two implementations and evade recognition half the time — a caveat we have not seen stated elsewhere.

D. Certificate-level fingerprints and identity exposure

Both TLS 1.2 stations present a three-certificate chain rooted in V2G Root CA G2 (Hubject) via CPO Sub1 CA G2.3 and CPO Sub2 CA G2.3.2: ECDSA P-256 with SHA-256, 90-day leaves, and an Online Certificate Status Protocol (OCSP)

TABLE III. VEHICLE-SIDE (EVCC) CLIENTHELLO FINGERPRINTS. “RESEARCH EVCC” IS THE TOOL USED FOR THE STATION CORPUS, SHOWN FOR COMPARISON.

EVCC	Suites offered	SNI	On bad chain
VW ID.3, ID.4	0xc023, 0xc025	none	unknown_ca (48)
Hyundai Ioniq 6	0xc023, 0xc025	none	silent FIN
Research EVCC	31 suites, TLS 1.3	illegal	—

responder at `ocsp.hubject.com:8080` — unreachable from a vehicle whose only link is the charging cable, so revocation cannot be checked at the point of use. Despite the shared CA the leaves differ by construction: the Tritium leaf has an 8-byte subject key identifier and an OU, the ABB leaf a conventional 20-byte SHA-1 identifier and none. Certificate tooling is thus an orthogonal fingerprint dimension that survives where TLS parameters converge.

Because TLS 1.2 sends the Certificate message in cleartext, both stations disclose their EVSE ID (`DE*IOY*E100400, DE*IOY*E224918`) and operator to any PLC listener before authentication; the TLS 1.3 station does not. This puts the EVSE ID in an unfortunate position: a vehicle cannot check whether the identity in a SECC certificate belongs to the charger it is plugged into, which is what makes the relay attack of [7] possible, while our traces show that same field broadcast in the clear. The identifier is simultaneously useless for authenticating the charger and perfectly serviceable for tracking it.

E. The other side of the connector: vehicle fingerprints

Of the 11 vehicles in the second corpus, three reach the TLS layer and yield ClientHellos (Table III); the rest negotiate DIN SPEC 70121 or plaintext ISO 15118-2. The two VW Group models share one JA3, as expected of a common EVCC platform, and the Hyundai is distinct — client-side fingerprinting separates vendor platforms here too, on far shorter hellos than any browser emits. Both send extension 3 (`trusted_ca_keys`), essentially unused on the web but natural in V2G.

Three observations matter more than the hashes. First, *both* production EVCCs offer 0xc025, the suite ISO 15118-2 mandates and that neither observed station selected: the mandatory suite is supported by the vehicles and declined by the infrastructure. Second, both reject our untrusted chain — the correct outcome — but signal it differently: the VW EVCCs send a fatal `unknown_ca` (48), the Hyundai sends no alert and closes the connection after 350 ms. RFC 5246 [17] requires an alert; silent teardown is non-conformant, harder to diagnose in the field, and a stable behavioural fingerprint in itself. Third, neither offers TLS 1.3. The TLS 1.3 station of Section IV-B is therefore unreachable in that mode by any vehicle in our sample, so its certificate-encryption privacy benefit does not accrue to these cars.

F. A conformance hazard, and a measurement caveat

The EVCC used for the station corpus places a link-local IPv6 literal with a zone identifier (`fe80::213:95ff:fe2e:f9dc%qca0`) in the Server

Name Indication (SNI) extension, which RFC 6066 forbids [18]. Two of the six tested stacks — GnuTLS and Java JSSE — abort with `illegal_parameter`; OpenSSL, mbedTLS, wolfSSL and Go ignore the field. The vehicle corpus bounds the scope: no production EVCC we observed sends SNI at all, so this is a hazard for measurement and test tooling rather than evidence that the fleet constrains SECC stack choice. It stays a live risk for any EVCC that does emit SNI, and it is why the 0xc023 selections of F1 and F2 should be read carefully — a research hello offering 31 suites and TLS 1.3 is not what a station meets in service. Comparing the two corpora quantifies that gap and strengthens the case for a fixed, published probe hello.

V. DISCUSSION AND LIMITATIONS

Fingerprints serve three audiences. For *regulators*, JA3S/JA4S with the SDP behaviour form a cheap passive probe reporting not only whether a station speaks TLS but how well — F2 would pass any pass/fail audit while negotiating none of the last decade’s record-layer hardening. For *operators*, they identify the stack and configuration of remote hardware without firmware access. For *attackers*, the same signal is reconnaissance, which is one more reason to prefer the ISO 15118-20 posture, where TLS is mandatory rather than negotiated and the certificate is encrypted. That posture is arriving by regulation: installations after 2027 must support ISO 15118-20 under the European Alternative Fuels Infrastructure Regulation [6], so the fleet will acquire TLS at scale without acquiring any means of inspecting how it is configured. A baseline established now is what makes that transition auditable — and attribution by CA will not substitute, since at least four commercial V2G PKIs serve this market [12] yet every chain we observed rooted in one of them.

Limitations. The station corpus is geographically concentrated and skewed towards one manufacturer; three completed handshakes and three vehicle ClientHellos are case studies establishing feasibility, not a survey. All station fingerprints were elicited by one research EVCC, and Section IV-F shows that hello differs materially from what production vehicles send, so a fleet-wide database needs a fixed, published probe hello — which the ISO 15118 conformance community is well placed to standardise. Our library covers six stacks in one distribution’s versions, so “matches OpenSSL 3.0.13” means “indistinguishable from”, not “is”.

VI. CONCLUSION AND FUTURE WORK

TLS in ISO 15118 is usually measured as a bit. Treating it as a fingerprint reveals more: two stations serve whichever security level the client requests, so availability figures depend on how they were measured; three working deployments run three demonstrably different stacks behind a single PKI; a station built in 2024 negotiates none of the preceding decade’s TLS 1.2 hardening; TLS 1.2 charging discloses the charge point’s identity in cleartext; and, on the other side of the connector, vehicles offer the mandatory suite the infrastructure

declines, disagree on how to refuse an untrusted certificate, and cannot yet reach the one TLS 1.3 station we found. The replay-based library behind these attributions is cheap to build, touches no live infrastructure and is straightforward to extend; we release our tooling to support a shared ISO 15118 fingerprint database.

Future work follows three strands. First, widening the library across stack versions and build options would turn “indistinguishable from OpenSSL 3.0.13” into a version estimate; the replay method makes this cheap, if tedious. Second, a fixed probe hello should be standardised so that fingerprints from different campaigns are comparable; the vehicle corpus gives a principled basis for choosing one that resembles a production EVCC rather than a test tool. Third, the station corpus should be widened geographically and across operators, and repeated after the 2027 deadline, so that the transition is measured rather than assumed.

ACKNOWLEDGMENT

This work was partially 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.

This work was partially created with the assistance of Claude Cowork. Only the data collection was done by a human.

REFERENCES

- [1] ISO, “ISO 15118-2:2014 — Road vehicles: Vehicle to grid communication interface — Part 2: Network and application protocol requirements,” 2014.
- [2] ISO, “ISO 15118-20:2022 — Part 20: 2nd generation network and application protocol requirements,” 2022.
- [3] DIN, “DIN SPEC 70121:2014 — Digital communication between a DC EV charging station and an electric vehicle,” 2014.
- [4] M. Szakály, S. Köhler, and I. Martinovic, “Current affairs: A security measurement study of CCS EV charging deployments,” arXiv:2404.06635, 2024.
- [5] M. Szakály, M. Strohmeier, I. Martinovic, and S. Köhler, “DCeption: Real-world wireless man-in-the-middle attacks against CCS EV charging,” arXiv:2601.15515, 2026.
- [6] J. Löw, L. Eder, A. Müller, and H.-J. Hof, “Market-analysis-driven methodology for assessing charging station cybersecurity,” arXiv:2605.22151, 2026.
- [7] J. Löw, V. Vasu, T. Hutzelmann, and H.-J. Hof, “Charge it to my neighbor: A relay attack on ISO 15118 plug and charge payment,” in *USENIX VehicleSec*, 2026.
- [8] L. Eder, J. Löw, and H.-J. Hof, “Charging communication sniffing and man-in-the-middle attacks,” in *ACM e-Energy*, 2025, doi:10.1145/3679240.3734648.
- [9] J. Löw, D. Bayerl, K. Mayer, and H.-J. Hof, “DrainDead: Emptying batteries of parked electric vehicles,” 2025.
- [10] J. Althouse, J. Atkinson, and J. Atkins, “TLS fingerprinting with JA3 and JA3S,” Salesforce Engineering, 2019.
- [11] J. Althouse et al., “JA4+ network fingerprinting suite,” FoxIO, 2023.
- [12] J. Löw, K. Mayer, and H.-J. Hof, “Fast charging communication and cybersecurity: A technology review,” CARISSMA Institute of Electric, Connected and Secure Mobility, Technische Hochschule Ingolstadt, 2024.
- [13] S. Köhler, R. Baker, M. Strohmeier, and I. Martinovic, “Broken-wire: Wireless disruption of CCS electric vehicle charging,” in *NDSS*, 2023.
- [14] M. Conti, D. Donadel, R. Poovendran, and F. Turrin, “EVEexchange: A relay attack on electric vehicle charging system,” in *ESORICS*, 2022.
- [15] N. AlFardan and K. Paterson, “Lucky Thirteen: Breaking the TLS and DTLS record protocols,” in *IEEE S&P*, 2013.
- [16] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Pironti, and P.-Y. Strub, “Triple handshakes and cookie cutters: Breaking and fixing authentication over TLS,” in *IEEE S&P*, 2014.
- [17] T. Dierks and E. Rescorla, “The TLS protocol version 1.2,” RFC 5246, 2008.
- [18] D. Eastlake, “TLS extensions: Extension definitions,” RFC 6066, 2011.
- [19] E. Rescorla et al., “TLS renegotiation indication extension,” RFC 5746, 2010.
- [20] K. Bhargavan et al., “TLS extended master secret extension,” RFC 7627, 2015.
- [21] P. Gutmann, “Encrypt-then-MAC for TLS and DTLS,” RFC 7366, 2014.