DOCKETED	
Docket Number:	22-EVI-06
Project Title:	Vehicle-Grid Integration
TN #:	256283
Document Title:	ISO 15118 Workshop - Tesla Plug and Charge Lite Presentation
Description:	N/A
Filer:	Jeffrey Lu
Organization:	California Energy Commission
Submitter Role:	Commission Staff
Submission Date:	5/10/2024 10:18:19 AM
Docketed Date:	5/10/2024

TESLA

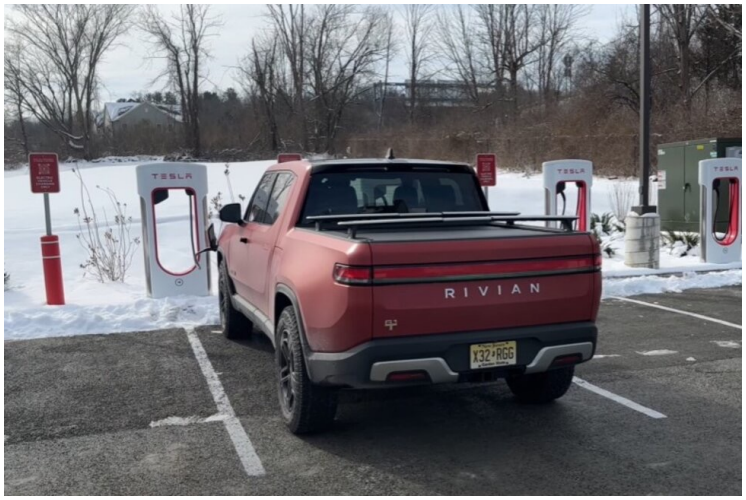
ISO 15118 Lite

Ideas for speeding up plug-to-charge latency

LAST EDITED
Apr 1, 2024

Background

Tesla is adopting ISO 15118 for our Supercharger network in North America. We have recently launched, and cars are using ISO 15118-2 PnC to charge on our network.

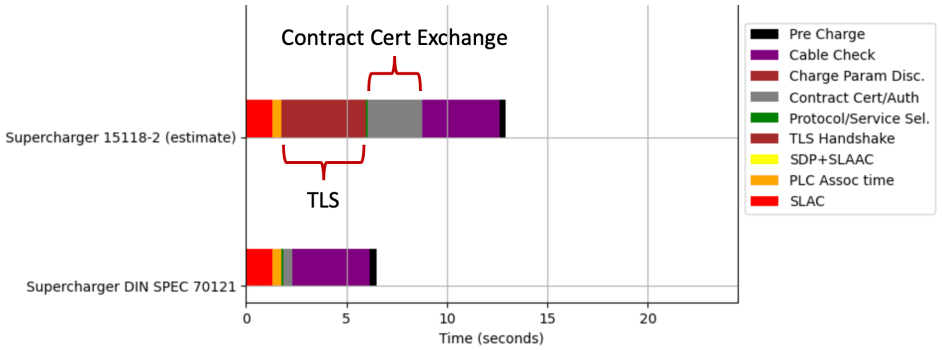


Background

DIN SPEC 70121 protocol takes around 6 seconds from plug in to start of charge, but ISO 15118-2 with Plug and Charge takes up to 12-20 seconds.

Plug to charge latency and reliability are important to EV owners.

For this reason, we'd like to discuss some potential amendments to the ISO 15118-2 protocol which have the aim of improving latency and reliability.



Optimization Idea #1

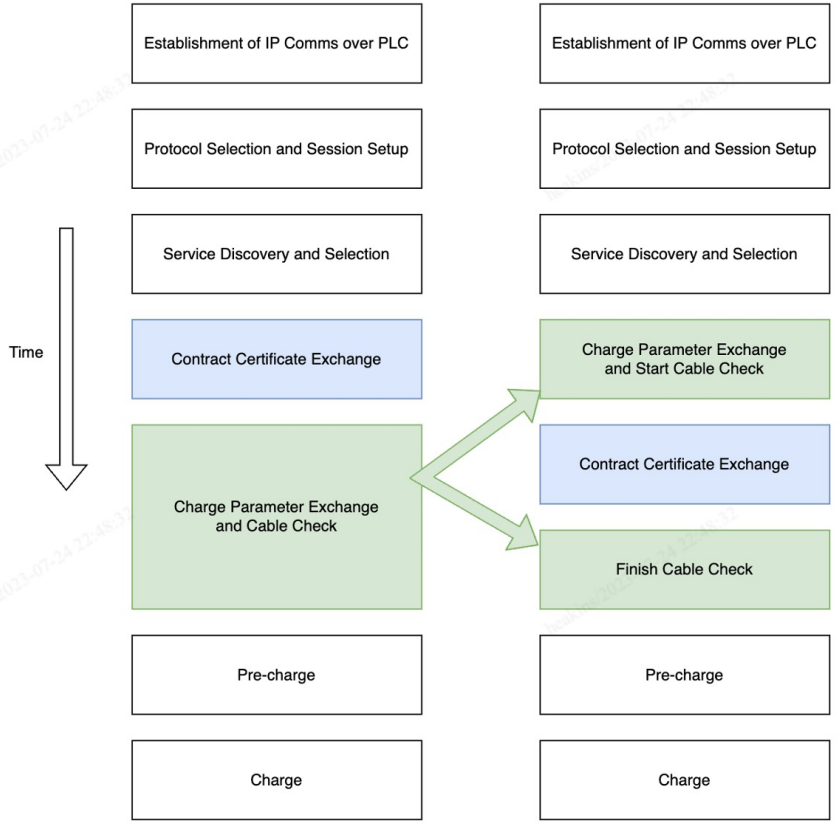
Do Cable Check in Parallel with CC Exchange

Cable check and contract cert-based authentication take a long time in ISO 15118 PnC, and there's no reason we can't do these in parallel…

Let's rearrange the messages and send the Charge Parameters before auth starts, allowing the EVSE to start cable check earlier.

Can be done in ISO 15118-2 or -20

ServiceID (unsignedshort)	ServiceName	ServiceCategory	Description
0			Reserved by ISO/IEC
1	AC_DC_Charging	EVCharging	All charging services as defined by SupportedEnergyTransferMode in subclause 8.5.2.3.
2	Certificate	ContractCertificate	Service allowing to update or install Contract Certificates.
3	InternetAccess	Internet	Service for standard protocols like HTTP, HTTPs, FTP, etc.
4	UseCaseInformation	EVSEInformation	Service enabling the exchange of use case specific information about the EVSE.
5 – 60000			Reserved by ISO/IEC
5000	Parallel CC and Cable Check	Protocol improvement	Selecting this service will enhance the communication timeup. See Section: TBD
60001 – 65535			Reserved for implementation specific use



Optimization Idea #2

TLS

What do we really care about?

- The EV cares that it's is going to be charged for the correct amount of energy.
- The EVSE/CPO cares that it's going to get paid for the energy being delivered.
- Safety

Impacts of TLS on charging in 15118-2

- Lower Reliability – failed sessions due to expired TLS certificates, or other compatibility issues due to complexity
- Limited Security – by only validating SECC TLS cert is from a trusted issuer, one charger's compromised private key compromises the entire region
- Slower – TLS handshake is one of the biggest factors in latency
- No impact on safety – ISO 15118 is not a safety-critical component of charging.

Optimization Idea #2

TLS

What about the other layer of security in 15118-2?

- Contract certificates and signed metering receipts.
- Very valuable – they give exactly what we care about.
- They don't rely on TLS for security
- Signed metering receipts are broken in 15118-2 but fixable.

Idea #1: simplify – focus on a single security layer and make it strong.

With just contract certificates, the CPO can be confident that they will be paid for energy delivered.

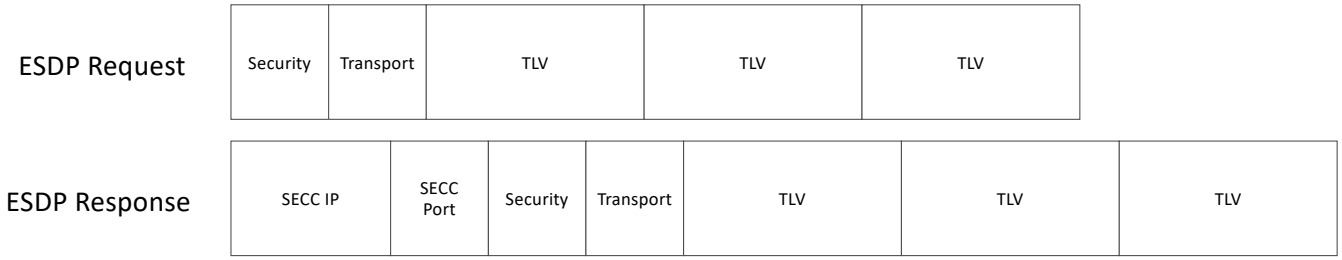
If we fix signed metering receipts, the vehicle can be confident that they are getting an agreed upon amount of energy, regardless of if TLS is used or not.

If we then remove TLS, we can improve reliability, performance and interoperability without sacrificing useful security.

Extended SECC Discovery Protocol (ESDP)

Required for dropping TLS in a backwards compatible way

- Provides a mechanism to exchange information at the earliest point of the charge session after the SLAC connection is established
- Includes regular SDP fields
- UDP packet sent by the EVCC to the IPv6 destination local-link multicast address (FF02:1) to a different port than regular SDP
- Shall be ignored by SECCs which do not support (and then the EVCC shall fall back to regular SDP)
- Extensible for future growth
- An extension could be used for the EVCC and SECC to negotiation PnC without TLS in a backwards compatible way.

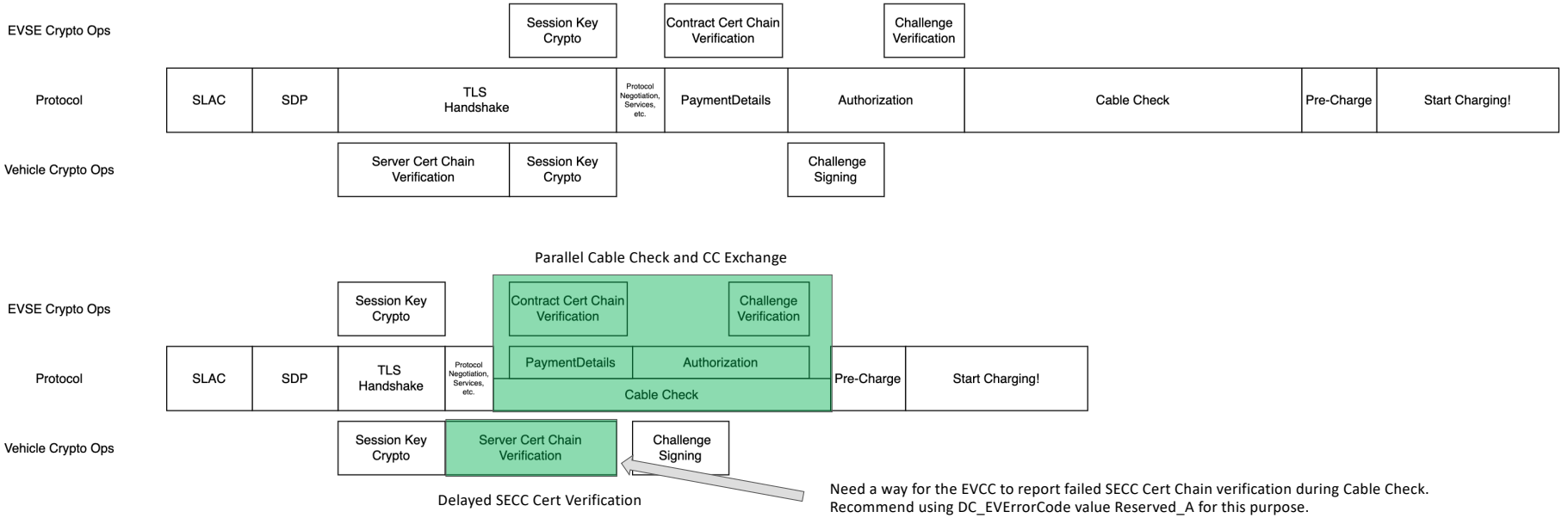


Optimization #3

Delayed SECC Cert Chain Verification

Instead of fully dropping TLS (which may be difficult in certain regions for compliance reasons), can we just perform less work in the handshake?

EVCC can complete the handshake before it has validated the cert chain and validate the certs in the background afterwards.



A look at Performance Impact

Example: A vehicle which takes 3s to implement check SECC cert chain verification, 1s to send contract cert chain, and 1s to sign challenge using contract cert private key.