QualityLogic

Collecting Timing Statistics Efficiently with the QualityLogic CCS Analyzer

A Case Study in CCS Charging Performance Analysis

DEKRA, a global leader in charging infrastructure testing and certification, constantly seeks to enhance its testing portfolio with tools that allow more sophisticated insights of charging session data. In particular, when testing charging infrastructure for reliability and interoperability under different scenarios, a large amount of valuable data is collected and analyzed to provide useful insights. Recently, DEKRA was commissioned by a major Charge Point Operator (CPO) to perform end-to-end testing with a number of different charge points and electric vehicles. The goal was to establish a baseline for interoperability and reliability that includes the EV to EVSE communication and the entire end-to-end communication.

The speed of the pairwise testing made real-time analysis of the charging sessions and end-to-end communication via OCPP and OCPI, respectively, challenging. The only real-time analysis that the schedule allowed was to investigate the issues that prevented charging from starting or completing. Even then, the time available for such analysis was limited, and numerous session files were saved for later analysis. The goal of the project was to make sure that as many pairwise tests were conducted as possible. So, most of the analysis happened post-test but on a compressed schedule. Given that the testing consisted of more than 100 pairs of test runs, and the analysis needed to cover both interoperability issues, if present, and up to 40 timing statistics for every test, the challenge DEKRA faced was how to accomplish the analysis more efficiently than traditional methods of such analysis.

The Traditional Approach to Charging Session Analysis

The current approach to charging session analysis is to use an engineering tool such as Keysight's Man-in-the-Middle Charge Discovery System or Iotecha's V2G Sniffer to capture and display the charging session in real-time. The systems enable real-time observation of the charging session but do minimal (if any) analysis of the session for interoperability or timings. The analysis of the real-time session data is done by an expert with deep knowledge of the charging standards, typical issues encountered, and the tools needed to capture and display charging session data.

For post-testing analysis, the capture/display systems export a PCAP or PCAPng file of the charging session. These charging session PCAP files are then displayed in a Wireshark open-source packet analyzer from the Wireshark Foundation. It enables the detailed examination of the message packets of a charging session by an expert, as well as the extraction of timing statistics from the packet analyzer.

While Wireshark provides a method to examine the contents of the

"As the performance of chargers and EV charging becomes more important, the ability to quickly extract timing statistics from charging session files will become even more useful."

~ Beat Kreuter, DEKRA

charging session interactions, it still requires an expert investigation to determine the causes of interoperability issues and to extract charging statistics (lesser expertise but still a manual process). The interoperability investigation an expert undertakes starts with examining the Wireshark packets at the end of the session, presumably where the issue occurred that caused the session to fail. Charging sessions could produce hundreds of message packets that need to be analyzed. The expert works backward through the packets in anywhere from 10 minutes to an hour or more, depending on complexity, to determine the cause of the failure.

Charging failures are typically caused by an anomaly, such as a SLAC failure, an incorrect session ID, or a mismatch in timing cycle expectations. Once the expert identifies the root cause, further investigation of the charging session packets is deemed unnecessary. The downsides of this process are that 1) other anomalies besides the root cause issue are probably not discovered, and 2) successful charging sessions are not investigated at all for non-fatal but problematic anomalies.

"This is a unique product and compliments the current set of CCS engineering tools and sniffers for charging session analysis."

~ Beat Kreuter, DEKRA

To extract timing statistics from a charging session, the engineer needs to find the starting packet of a specific timing parameter—e.g., Session Setup time—extract the time stamp of that packet, and enter it in a spreadsheet. They next need to find the ending packet, extract the time stamp, and enter it into the spreadsheet. While this is relatively easy for a single timing parameter, it may take 20 minutes or so to extract 40 parameters and log them in a spreadsheet.

For someone doing timing analysis infrequently, this is not a major time sink. However, for 10, 20, or more timings and multiple charging sessions to be analyzed, this becomes a time-consuming process. For the DEKRA Interop testing project, over 100 charging sessions are needed.

Automating PCAP File Analysis

Beat Kreuter, VP of Business Line Product Safety Testing and a key contributor to the formal testing processes and documents of CharIN, recognized the value of the QualityLogic CCS Analyzer while still in the prototype stage. Beat was an early user, but he found that the Analyzer was most valuable to him when analyzing the timing statistics for DEKRA's interoperability and end-to-end testing project. The tool provided a set of unique values:

- It was simple to use, didn't require hardware, and made it easy to view and extract the charging session data he needed for the project.
- The timing statistics of the CCS Analyzer were particularly useful and greatly improved the efficiency of extracting the timings of the charging sessions from the PCAP files. There is no other tool that provides similar utility.
- The session chart was also valuable. Similar graphics are available in other tools, but the QualityLogic tool makes it very easy to view and manipulate the graph of the charging sessions.
- The anomaly identification both reduces the time to analyze interop failures and, in many instances, confirms the real-time analysis. The tool greatly enhances the manual process of failure identification typically conducted by an expert.

According to Beat, "QualityLogic came up with the concept of a tool that would automate the analysis of charging session PCAP files over a year ago and asked me to advise them. It's been rewarding to see a new product develop so rapidly to become a very useful tool to advance charging interoperability and performance. I found the timing statistics and graphs

particularly valuable as they saved me time manually collecting them from the PCAPs. This is a unique product and compliments the current set of CCS engineering tools and sniffers for charging session analysis."

The bottom line for DEKRA was the ability to extract essential charging session performance key indicators, such as timings, in a matter of minutes instead of hours.

Automating Interoperability Issue Investigation

According to Beat, the automation of charging session analysis brings a number of benefits. It speeds up the analysis an engineer must do while doing a more thorough analysis of the session file in less time. The CCS Analyzer identifies anomalies and performance issues that may not cause a failed charging session but are nonetheless valuable information for improving interoperability. These additional issues are often challenging to find and may be buried deep in the session file. Once a primary cause of a failed session has been identified, the engineer will stop analyzing the file and move on to other high-priority tasks rather than continuing the analysis to identify these "buried" anomalies.

"The industry needs such a tool, and we are glad QualityLogic invested in its development," says Beat. "It complements the other tools in my session analysis tool kit, and the fact that it works with PCAP files from any system (as opposed to proprietary hardware) makes it even more valuable for the industry."

Automating Timings Extraction

"When we used the CCS Analyzer to extract timings statistics from many files, it only had about a dozen specific timings. But even that was a valuable time-saver for us. QualityLogic has since added another 30+ timings, making the tool even more valuable," says Beat. **"As the performance of chargers and EV charging becomes more important, the ability to quickly extract timing statistics from charging session files will become even more useful."**

TimingStatsTable

Phase	Duration (s)	Start (frame, time)	End (frame, time)	Notes
CM_SLAC_PARM.REQ to CM_SLAC_PARM.CNF	0.004	CM_SLAC_PARM.REQ (16, 10.270)	CM_SLAC_PARM.CNF (17, 10.274)	
CM_SLAC_PARM.REQ to CM_SLAC_MATCH.CNF	1.344	CM_SLAC_PARM.REQ (16, 10.270)	CM_SLAC_MATCH.CNF (36, 11.614)	
SDP_Req to SDP_Res	0.025	SDPRequest (48, 16.710)	SDPResponse (51, 16.735)	
SessionSetupReq to SessionSetupRes	0.070	SessionSetupReq (84, 18.411)	SessionSetupRes (87, 18.482)	
ChargeParameterDiscoveryReq to Res - finished	1.820	ChargeParameterDiscoveryReq (152, 25.7...	ChargeParameterDiscoveryRes (165, 27.543)	
CableCheckReq to CableCheckRes - finished	20.240	CableCheckReq (170, 27.772)	CableCheckRes (469, 48.012)	
PreChargeReq to PreChargeRes - finished < 500 V	4.518	PreChargeReq (472, 48.084)	PreChargeRes (540, 52.602)	EV Target Voltage: 424.5 V
PowerDeliveryReq to PowerDeliveryRes	0.039	PowerDeliveryReq (543, 52.692)	PowerDeliveryRes (545, 52.732)	
AuthorizationReq to AuthorizationRes - finished (OCPP and Backend Performan...	5.479	AuthorizationReq (108, 20.152)	AuthorizationRes (149, 25.632)	Authorization: PnC
Req/Res cycle (average from at least 4 Req/Res)	0.024			
PnC only - SDPRes to PaymentDetailRes	2.557	SDPResponse (51, 16.735)	PaymentDetailsRes (105, 19.292)	
PnC only - PaymentDetailReq to AuthorizationRes	1.579	PaymentDetailsReq (103, 18.782)	AuthorizationRes (110, 20.361)	
B2 to CM_SLAC_PARM.REQ	0.093	ControlPilotMessage (15, 10.177)	CM_SLAC_PARM.REQ (16, 10.270)	
CM_SLAC_PARM.REQ to CM_START_ATTEN_CHAR.IND	0.110	CM_SLAC_PARM.REQ (16, 10.270)	CM_START_ATTEN_CHAR.IND (19, 10.3...	
CM_MNBC_SOUND.IND Count 9 - 0	0.272	CM_MNBC_SOUND.IND (22, 10.470)	CM_MNBC_SOUND.IND (31, 10.742)	
CM_SLAC_MATCH.REQ/CNF to SDPReq	5.096	CM_SLAC_MATCH.CNF (36, 11.614)	SDPRequest (48, 16.710)	
SDP_Req to SupportedAppProtocolReq	1.641	SDPRequest (48, 16.710)	SupportedAppProtocolReq (80, 18.351)	
SDP_Req to SessionSetupReq	1.701	SDPRequest (48, 16.710)	SessionSetupReq (84, 18.411)	
AuthorizationRes to ChargeParameterDiscoveryReq	0.091	AuthorizationRes (149, 25.632)	ChargeParameterDiscoveryReq (152, 25.722)	
ChargeParameterDiscoveryRes to CableCheckReq (State B to C)	0.229	ChargeParameterDiscoveryRes (165, 27.5...	CableCheckReq (170, 27.772)	
CableCheckRes to PreChargeReq	0.072	CableCheckRes (469, 48.012)	PreChargeReq (472, 48.084)	
PreChargeRes to PowerDeliveryReq	0.090	PreChargeRes (540, 52.602)	PowerDeliveryReq (543, 52.692)	
PowerDeliveryRes to CurrentDemandReq	0.072	PowerDeliveryRes (545, 52.732)	CurrentDemandReq (548, 52.804)	
EVCC Ready to Charge (B2 to PowerDeliveryReq - Authorization)	37.036	ControlPilotMessage (15, 10.177)	PowerDeliveryReq (543, 52.692)	
Res/Req cycle time (average from at least 4 Res/Req)	0.350			
PnC only - SDPRes to AuthorizationReq	3.417	SDPResponse (51, 16.735)	AuthorizationReq (108, 20.152)	
SDP_Res to SupportedAppProtocolReq	1.616	SDPResponse (51, 16.735)	SupportedAppProtocolReq (80, 18.351)	
Overall Charging	75.273	SessionSetupReq (84, 18.411)	SessionStopReq (983, 93.685)	
Charge Session Setup	34.342	SupportedAppProtocolReq (80, 18.351)	PowerDeliveryReq (543, 52.692)	
Charge Loop (DC)	38.919	PowerDeliveryReq (543, 52.692)	PowerDeliveryRes (945, 91.612)	RemainingTimeToFullSoC decreased by 60 (s)
Welding Detection Loop	1.717	WeldingDetectionReq (951, 91.784)	WeldingDetectionRes (980, 93.502)	

exceeded_limits.pcapng (session: 1)

Figure 1. Timing statistics can offer additional insights

About QualityLogic's CCS Analyzer

QualityLogic's CCS Analyzer is a new class of tool for analysis of Combined Charging System (CCS) traffic between the EV and the EVSE. The CCS Analyzer is an "expert system" that improves productivity of human experts by automating the analyses the session traffic to identify anomalies that cause failed charging sessions, extract timing data from the files and uncover non-fatal anomalies that may cause future issues. The tool is hardware agnostic and compliments existing engineering test tools and sniffers.

For a demonstration or to learn more, contact us at info@qualitylogic.com