



OCPP 2.0.1 Edition 4
Errata 2026-04

Table of Contents

Disclaimer	1
Scope	2
Terminology and Conventions	2
0. Part 0	3
1. Part 1	4
2. Part 2	5
2.1. Page 12 - (2026-02) - Updated reference 17 and added new references	5
2.2. Page 21 - (2026-02) - Inclusion of RFC 8441 for HTTP/2	5
2.3. Page 21 - (2026-02) - A00.FR.308 - Additional requirement references	5
2.4. Page 23 - (2026-02) - A00.FR.403 - Additional requirement references	6
2.5. Page 74 - (2026-04) - B12 - Reset with ongoing transaction allows wait for cable disconnection.	6
2.6. Page 24 - (2026-02) - A00.FR.411 - Additional requirement references	8
2.7. Page 34 - (2026-02) - A02.FR.02 - Requirement reference correction	8
2.8. Page 37 - (2026-02) - A03.FR.02 - Requirement reference correction	8
2.9. Page 35 - (2026-02) - A02.FR.19 conditions made more specific [1158]	8
2.10. Page 37 - (2026-02) - A03.FR.02 Only applies in security profile 3 [1167].	9
2.11. Page 224 - (2026-02) - I06.FR.03 - added note about privacy [715]	9
2.12. Page 293 - (2026-04) - L01.FR.04 - removed ECSchnorr algorithm option and clarify the signature verification the Charging Station must do	9
2.13. Page 412 - (2026-02) - Improved description of UnitOfMeasureType [670]	10
3. Part 3	11
4. Part 4	12
4.1. Page 10 - (2026-02) - Update handling of unknown message type [1163]	12
4.2. Page 16 - (2026-02) - MessageTypeNotSupported is deprecated [1161]	12
4.3. Page 16 - (2026-02) - Section 4.4 Extension fallback mechanism removed [1163].	12
5. Part 5	13
6. Part 6	14
6.1. General.	14
6.2. Charging Station.	15
6.2.1. Page 23 - (2026-02) - TC_A_20_CS - Added note that the testcase will end when the SetNetworkProfile is rejected . .	15
6.2.2. Page 57 - (2026-04) - TC_B_21_CS - Clarification of charging station reboot behaviour with ongoing transactions. . .	15
6.2.3. Page 65 - (2026-04) - TC_B_41_CS - Clarification of charging station reboot behaviour with ongoing transactions. . .	16
6.2.4. Page 107 - (2026-02) - TC_C_17_CS - Corrected wrong triggerReason [1113]	18
6.2.5. Page 124 - (2026-02) - TC_C_38_CS - Removed memory state IdTokenCached [1164]	19
6.2.6. Page 150/151 - (2026-02) - TC_C_26_CS - Clarification of main test and tool validation steps.	19
6.2.7. Page 401 - (2026-02) - TC_L_07_CS - Path to non-existent firmware updated	20
6.2.8. Page 449 - (2026-02) - TC_M_24_CS - GetCertificateStatusRequest is not intended to be sent for the V2G leaf certificate	21
6.2.9. Page 552 - (2026-02) - TC_O_24_CS - Additions to tool validations	21
6.2.10. Page 585 - (2026-02) - Reusable state Booted now accepts StatusNotification Unavailable [1162].	22
6.3. CSMS	24

Disclaimer

Copyright © 2010 – 2026 Open Charge Alliance. All rights reserved.

This document is made available under the **Creative Commons Attribution-NoDerivatives 4.0 International Public License** (<https://creativecommons.org/licenses/by-nd/4.0/legalcode>).

Version History

Version	Date	Description
2026-04	2026-05-06	Includes errata for Part 2, 4, 5 and 6 of OCPP 2.0.1 Edition 4.
2026-02	2026-03-11	Includes errata for Part 2, 4, 5 and 6 of OCPP 2.0.1 Edition 4.
2025-11	2025-12-03	Includes errata for Part 1, 2, 4, 5 and 6 of OCPP 2.0.1 Edition 3.
2025-09	2025-09-30	Includes errata for Part 1, 2, 4, 5 and 6 of OCPP 2.0.1 Edition 3.
2025-06	2025-07-08	Includes errata for Part 2, 5 and 6 of OCPP 2.0.1 Edition 3.
2025-04	2025-04-30	Includes errata for Part 2, 5 and 6 of OCPP 2.0.1 Edition 3.
2025-02	2025-03-06	Includes errata for Part 2, 5 and 6 of OCPP 2.0.1 Edition 3.
2025-01	2025-01-23	Includes errata for Part 1-4 of OCPP 2.0.1 Edition 3
2024-11	2024-11-14	Includes errata for Part 5 and Part 6 of OCPP 2.0.1 Edition 3
2024-09	2024-09-25	Includes errata for Part 4, Part 5 and Part 6 of OCPP 2.0.1 Edition 3
2024-06	2024-06-27	Includes errata for Part 5 and Part 6.

Scope

This document contains errata on the OCPP 2.0.1 documentation. These errata have to be read as an addition to the release of OCPP 2.0.1 Edition 4.

The errata do not affect any schemas of OCPP messages. Certain errata do contain changes to requirements or even new requirements, but only in cases where a requirement contains an obvious error and would not or could not be implemented literally. New requirements are only added when they were already implicitly there. These changes have been discussed in or were proposed by the Technology Working Group of the Open Charge Alliance.

The appendices of the OCPP specification can be updated without requiring a new OCPP release. This mainly concerns the components and variables of the OCPP device model, which can be extended with new components or variables, as long as they are optional.

Terminology and Conventions

Bold: when needed to clarify differences, bold text might be used.

The errata entries are sorted by page number of the affected section of the specification document. When an errata entry affects multiple parts of the specification, then the various changes are grouped together with subsections referring to the pages affected by those changes.

This is version 2026-04 of the errata. The errata of this version are marked with "(2026-04)" in the section title.

In some cases the issue number by which it was reported, is added in square brackets at the end of the section title, e.g. "[349]". For retrieval of the issue in the issue tracking system prefix the number with "OCPP20M", like "[OCPP20M-349]".

0. Part 0

Currently no new errata for OCPP 2.0.1 Edition 4 part 0.

1. Part 1

Currently no new errata for OCPP 2.0.1 Edition 4 part 1.

2. Part 2

2.1. Page 12 - (2026-02) - Updated reference 17 and added new references

Special Publication 800-57 updated with latest revision

	Reference	Description	Note
Old	[17]	National Institute of Standards and Technology. Special Publication 800-57 Part 1 Rev. 4, Recommendation for Key Management. January 2016. https://csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-4/final	
New	[17]	National Institute of Standards and Technology. Special Publication 800-57 Part 1 Rev. 5, Recommendation for Key Management. May 2020. https://csrc.nist.gov/pubs/sp/800/57/pt1/r5/final	

RFC 6818 defines certificate validation rules in A00.FR.308, A00.FR.403, and A00.FR.413.

Reference	Description
[26]	RFC 6818. Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. https://tools.ietf.org/html/rfc6818
[27]	European Cybersecurity Certification Group Sub-group on Cryptography, Agreed Cryptographic Mechanisms, Version 2.0, April 2025 https://certification.enisa.europa.eu/document/download/a845662b-aea0-484e-9191-890c4cfa7aaa_en?filename=ECCG%20Agreed%20Cryptographic%20Mechanisms%20version%202.pdf

2.2. Page 21 - (2026-02) - Inclusion of RFC 8441 for HTTP/2

No.	Type	Description
1	Name	TLS with Basic Authentication
...		
7	Remark(s)	<p>TLS allows a number of configurations, not all of which provide sufficient security. The requirements below describe the configurations allowed for OCPP.</p> <p>The Charging Station should include the same header as used in Basic Auth RFC 2617, while requesting to upgrade the http connection to a websocket connection as described in RFC 6455 and RFC 8441 (HTTP/2). The server first needs to validate the Authorization header before upgrading the connection.</p>

2.3. Page 21 - (2026-02) - A00.FR.308 - Additional requirement references

	ID	Precondition	Requirement definition	Note
Old	A00.FR.308		The Charging Station SHALL verify the certification path of the CSMS's certificate according to the path validation rules established in Section 6 of [3].	
New	A00.FR.308		The Charging Station SHALL verify the certification path of the CSMS's certificate according to the path validation rules established in Section 6 of [3] and Section 4 of [26]	

2.4. Page 23 - (2026-02) - A00.FR.403 - Additional requirement references

	ID	Precondition	Requirement definition	Note
Old	A00.FR.403		The CSMS SHALL verify the certification path of the Charging Station's certificate according to the path validation rules established in Section 6 of [3]	
New	A00.FR.403		The CSMS SHALL verify the certification path of the Charging Station's certificate according to the path validation rules established in Section 6 of [3] and Section 4 of [26]	

2.5. Page 74 - (2026-04) - B12 - Reset with ongoing transaction allows wait for cable disconnection

New requirements have been added to clarify the expected behavior when a reset is requested during an ongoing transaction, including the requirement to allow waiting for cable disconnection before performing the reset. Requirements were further updated for clarity.

New requirement

ID	Precondition	Requirement definition	Note
B12.FR.11	B12.FR.03 AND The transaction(s) on the EVSE(s) are terminated	The Charging Station MAY wait for cable disconnection before initiating a reboot	
B12.FR.12	B12.FR.07 AND The transaction on the EVSE is terminated	The Charging Station MAY wait for cable disconnection before initiating a reset of the EVSE	
B12.FR.13	B12.FR.11	The Charging Station SHALL reboot	
B12.FR.14	B12.FR.12	The Charging Station SHALL reset the EVSE	

Changed requirement

	ID	Precondition	Requirement definition	Note
Old	B12.FR.01	When the Charging Station receives a ResetRequest(OnIdle) AND a transaction is ongoing	The Charging Station SHALL respond with a ResetResponse(Scheduled) , to indicate whether the Charging Station will attempt to reset itself or EVSE after all transactions on Charging Station or EVSE have ended.	
New	B12.FR.01	When the Charging Station receives a ResetRequest(OnIdle) AND a transaction is ongoing	The Charging Station SHALL respond with a ResetResponse(Scheduled)	
Old	B12.FR.02	When the Charging Station receives a ResetRequest(Immediate) AND a transaction is ongoing	The Charging Station SHALL respond with a ResetResponse(Accepted) , to indicate whether the Charging Station will attempt to reset itself or EVSE.	
New	B12.FR.02	When the Charging Station receives a ResetRequest(Immediate) AND a transaction is ongoing	The Charging Station SHALL respond with a ResetResponse(Accepted)	

	ID	Precondition	Requirement definition	Note
Old	B12.FR.03	If no <i>evseld</i> is supplied AND If any transaction is in progress and an OnIdle reset is received.	The transaction of the Charging Station SHALL be terminated normally, before the reboot, e.g. as in E06 - Stop Transaction .	
New	B12.FR.03	B12.FR.01 AND If no <i>evseld</i> is supplied	The transaction of the Charging Station SHALL be terminated normally, before the reboot, e.g. as in E06 - Stop Transaction .	
Old	B12.FR.04	If no <i>evseld</i> is supplied AND If any transaction is in progress and an Immediate Reset is received.	The Charging Station SHALL attempt to terminate any transaction in progress and send a TransactionEventRequest (eventType = Ended) message with <i>triggerReason</i> = <i>ResetCommand</i> and <i>transactionInfo.stoppedReason</i> = <i>ImmediateReset</i> for each terminated transaction before performing a reboot.	
New	B12.FR.04	B12.FR.02 AND If no <i>evseld</i> is supplied	The Charging Station SHALL attempt to terminate any transaction in progress and send a TransactionEventRequest (eventType = Ended) message with <i>triggerReason</i> = <i>ResetCommand</i> and <i>transactionInfo.stoppedReason</i> = <i>ImmediateReset</i> for each terminated transaction before performing a reboot.	
Old	B12.FR.05	If an Immediate Reset without <i>evseld</i> is received and the TransactionEventResponse is not received within timeout.	The Charging Station SHALL queue the TransactionEventRequest , reboot and resend the TransactionEventRequest after the reboot.	
New	B12.FR.05	B12.FR.04 AND the TransactionEventResponse is not received within timeout.	The Charging Station SHALL queue the TransactionEventRequest , reboot and resend the TransactionEventRequest after the reboot.	
Old	B12.FR.07	If an <i>evseld</i> is supplied AND If a transaction is in progress on the EVSE and an OnIdle reset is received.	The transaction on the EVSE SHALL be terminated normally, before the reset, e.g. as in E06 - Stop Transaction .	
New	B12.FR.07	B12.FR.01 AND If an <i>evseld</i> is supplied	The transaction on the EVSE SHALL be terminated normally, before the reset, e.g. as in E06 - Stop Transaction .	
Old	B12.FR.08	If an <i>evseld</i> is supplied AND If a transaction is in progress on the EVSE and an Immediate Reset is received.	The Charging Station SHALL attempt to terminate the transaction in progress on the EVSE and send a TransactionEventRequest (eventType = Ended) message with <i>triggerReason</i> = <i>ResetCommand</i> and <i>transactionInfo.stoppedReason</i> = <i>ImmediateReset</i> before resetting the EVSE.	
New	B12.FR.08	B12.FR.02 AND If an <i>evseld</i> is supplied	The Charging Station SHALL attempt to terminate the transaction in progress on the EVSE and send a TransactionEventRequest (eventType = Ended) message with <i>triggerReason</i> = <i>ResetCommand</i> and <i>transactionInfo.stoppedReason</i> = <i>ImmediateReset</i> before resetting the EVSE.	

2.6. Page 24 - (2026-02) - A00.FR.411 - Additional requirement references

	ID	Precondition	Requirement definition	Note
Old	A00.FR.411		The Charging Station SHALL verify the certification path of the CSMS's certificate according to the path validation rules established in Section 6 of [3]. (Same as A00.FR.308)	
New	A00.FR.411		The Charging Station SHALL verify the certification path of the CSMS's certificate according to the path validation rules established in Section 6 of [3] and Section 4 of [26]. (Same as A00.FR.308)	

2.7. Page 34 - (2026-02) - A02.FR.02 - Requirement reference correction

	ID	Precondition	Requirement definition	Note
Old	A02.FR.02		The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 4.2.1.3 of [16].	
New	A02.FR.02		The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 8.1 of [27].	

2.8. Page 37 - (2026-02) - A03.FR.02 - Requirement reference correction

	ID	Precondition	Requirement definition	Note
Old	A03.FR.02		The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 4.2.1.3 of [16].	
New	A03.FR.02		The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 8.1 of [27].	

2.9. Page 35 - (2026-02) - A02.FR.19 conditions made more specific [1158]

The condition "maximum number of increments reached" in A02.FR.19 has been made more explicit by referring to CertSigningRepeatTimes configuration variable.

	ID	Precondition	Requirement definition	Note
Old	A02.FR.19	A02.FR.18 AND The maximum number of increments is reached	The Charging Station SHALL stop resending the SignCertificateRequest, until it is requested by the CSMS via a TriggerMessageRequest for SignChargingStationCertificate, SignV2GCertificate or SignCombinedCertificate.	

	ID	Precondition	Requirement definition	Note
New	A02.FR.19	A02.FR.18 AND CertSigningRepeatTimes is reached	The Charging Station SHALL stop resending the SignCertificateRequest , until it is requested by the CSMS via a TriggerMessageRequest for SignChargingStationCertificate, SignV2GCertificate or SignCombinedCertificate.	

2.10. Page 37 - (2026-02) - A03.FR.02 Only applies in security profile 3 [1167]

Creation of a new client certificate must only be done while operating with security profile 3, otherwise CSMS might refuse to sign a client certificate.

Changed requirement

	ID	Precondition	Requirement definition	Note
Old	A03.FR.02	When the Charging Station detects that the current Charging Station certificate will expire in one month.	The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 4.2.1.3 of [16].	
New	A03.FR.02	When the Charging Station is connected with security profile 3 AND detects that the current Charging Station certificate will expire in one month.	The Charging Station SHALL generate a new public / private key pair using one of the key generation functions described in Section 4.2.1.3 of [16].	

2.11. Page 224 - (2026-02) - I06.FR.03 - added note about privacy [715]

A note has been added about protecting privacy when showing messages.

	ID.	Precondition	Requirements
Old	I06.FR.03	I06.FR.02	The Charging Station SHALL display the updated tariff information to the EV Driver.
New	I06.FR.03	I06.FR.02	The Charging Station SHALL display the updated tariff information to the EV Driver. Note: Charging Station should take measures to protect the privacy of the EV Driver, for example by ensuring that only the authenticated user can view the message.

2.12. Page 293 - (2026-04) - L01.FR.04 - removed ECSchnorr algorithm option and clarify the signature verification the Charging Station must do

	ID.	Precondition	Requirements	Note
Old	L01.FR.04	When the Charging Station has successfully downloaded the new firmware	The signature SHALL be validated, by calculating the signature over the entire firmware file using the RSA-PSS or ECSchnorr algorithm for signing, and the SHA256 algorithm for calculating hash values.	

	ID.	Precondition	Requirements	Note
New	L01.FR.04	When the Charging Station has successfully downloaded the new firmware	The charging station SHALL verify the signature from the UpdateFirmwareRequest with the hash over the entire received firmware file computed using the signature hash algorithm.	The RSA-PSS or ECDSA algorithm is used for signing. Please refer to section [certificate_properties] for more details about signature algorithms and computing hash values.

2.13. Page 412 - (2026-02) - Improved description of UnitOfMeasureType [670]

Field Name	Field Type	Card.	Description
unit	string[0..20]	0..1	Optional. Unit of the value. Default = "Wh" if the (default) measurand is an "Energy" type. This field SHALL use a value from the list Standardized Units of Measurements in Part 2 Appendices. If an applicable unit is available in that list, otherwise a "custom" unit might be used.
multiplier	integer	0..1	Optional. Multiplier, this value represents the exponent to base 10. I.e. multiplier 3 means 10 raised to the 3rd power. Default is 0. The <i>multiplier</i> only multiplies the value of the measurand. It does not specify a conversion between units, for example, from kW to W.

3. Part 3

Currently no new errata for OCPP 2.0.1 part 3.

4. Part 4

4.1. Page 10 - (2026-02) - Update handling of unknown message type [1163]

The sentence about ignoring unknown message types has been improved slightly. It mentioned to ignore the message payload, but the intention was to ignore the entire message.

Old paragraph	When a system receives a message with a Message Type Number not in this list, it SHALL ignore the message payload. Each message type may have additional required fields.
New paragraph	When a system receives a message with a Message Type Number not in this list, it SHALL ignore the message payload . Each message type may have additional required fields.

4.2. Page 16 - (2026-02) - MessageTypeNotSupported is deprecated [1161]

As of OCPP 2.1 a message type number that is not supported is silently ignored, as described in section 4.4. This is therefore no longer a valid error code.

Valid Error Codes

ErrorCode	Description
...	...
MessageTypeNotSupported	(deprecated) A message with a Message Type Number that is not supported by this implementation.
...	...

4.3. Page 16 - (2026-02) - Section 4.4 Extension fallback mechanism removed [1163]

Section 4.4 "Extension fallback mechanism" is contradictory to section 4.1, which states to ignore unknown message types. It has therefore been removed.

5. Part 5

Currently no new errata for OCPP 2.0.1 Edition 4 part 5.

6. Part 6

6.1. General

Currently no errata for Part 6 General.

6.2. Charging Station

6.2.1. Page 23 - (2026-02) - TC_A_20_CS - Added note that the testcase will end when the SetNetworkProfile is rejected

Test case name	Upgrade Charging Station Security Profile - No valid CSMSRootCertificate installed
Test case Id	TC_A_20_CS
...	

Main (Test scenario)	
Charging Station	CSMS
2. The Charging Station responds with a SetNetworkProfileResponse	1. The Test System sends a SetNetworkProfileRequest with - configurationSlot is <Configured configurationSlot2> or <Configured configurationSlot> (the one currently not used for the active connection) - connectionData.messageTimeout <Configured messageTimeout2> - connectionData.ocppCsmsUrl <ocppCsmsUrl that is not currently active> - connectionData.ocppInterface <Configured ocppInterface2> - connectionData.ocppVersion OCPP20 - connectionData.securityProfile <Configured securityProfile2>
Note: If the Charging Station responds with a SetNetworkProfileResponse with status Rejected, then step 3/4 will not be executed.	
4. The Charging Station responds with a SetVariablesResponse	3. The Test System sends a SetVariablesRequest with variable.name is "NetworkConfigurationPriority" component.name is "OCPPCommCtrlr" attributeValue is <configurationSlot set at step 1>,<previous configurationSlot>

6.2.2. Page 57 - (2026-04) - TC_B_21_CS - Clarification of charging station reboot behaviour with ongoing transactions

The test case has been updated to provide clarity on when the charging station is allowed to reboot. The charging station is expected to reboot either after the transaction has ended or if the transaction ends before the cable is disconnected, once the cable is disconnected.

Test case name	Reset Charging Station - With Ongoing Transaction - OnIdle
Test case Id	TC_B_21_CS

Before (Preparations)
Configuration State: N/a
Memory State: N/a
Reusable State(s): State is <i>EnergyTransferStarted</i>

Main (Test scenario)	
Charging Station	CSMS
2. The Charging Station responds with a ResetResponse	1. The Test System sends a ResetRequest with type OnIdle
3. Execute Reusable State <i>StopAuthorized</i>	

Main (Test scenario)
<p>Notes(s): If the TxStopPoint contains EnergyTransfer, PowerPathClosed, or Authorized: The test system will wait the <Configured Long Operation Time Out> to determine if the charging station resets because the transaction has ended.</p> <p>If the charging station resets proceed to step 7. Else proceed with step 4.</p> <p>4. Execute Reusable State <i>EVConnectedPostSession</i></p> <p>5. Execute Reusable State <i>EVDisconnected</i></p> <p>Notes(s): Steps 4 and 5 will only be executed if TxStartPoint does not contain: EnergyTransferStarted, DataSigned, PowerPathClosed, or Authorized</p> <p>Notes(s): Step 6 will only be executed if TxStartPoint does not contain: EnergyTransferStarted, DataSigned, PowerPathClosed, Authorized, or EVConnected is ParkingBayOccupancy</p> <p>...</p>

Tool validations
<p>* Step 2: Message ResetResponse - status Scheduled</p> <p>* Step 7: Message BootNotificationRequest - reason ScheduledReset</p> <p>* Step 9: Message: StatusNotificationRequest - If the transaction was stopped charging station reset at step 3, then – for the connector involved in the transaction: connectorStatus Occupied – for other connectors of this EVSE: connectorStatus Available or Unavailable Else connectorStatus Available Message: NotifyEventRequest - eventData[0].trigger Delta - eventData[0].component.name "Connector" - eventData[0].variable.name "AvailabilityState" - If the transaction was stopped charging station reset at step 3, then – for the connector involved in the transaction: eventData[0].actualValue Occupied – for other connectors of this EVSE: eventData[0].actualValue Available or Unavailable - Else eventData[0].actualValue "Available"</p> <p>* Step 11: Message: SecurityEventNotificationRequest - type StartupOfTheDevice or ResetOrReboot</p> <p>Post scenario validations: - A message to report the state of a connector has been received for all connectors.</p>

6.2.3. Page 65 - (2026-04) - TC_B_41_CS - Clarification of charging station reboot behaviour with ongoing transactions

The test case has been updated to provide clarity on when the charging station is allowed to reboot. The charging station is expected to reboot either after the transaction has ended or if the transaction ends before the cable is disconnected, once the cable is disconnected.

Test case name	Reset Charging Station - With multiple ongoing transactions - OnIdle
Test case Id	TC_B_41_CS

Main (Test scenario)	
Charging Station	CSMS

Main (Test scenario)	
2. The Charging Station responds with a ResetResponse	1. The Test System sends a ResetRequest with type OnIdle
3. Execute Reusable State <i>StopAuthorized</i> for EVSE.id = 1	
4. Execute Reusable State <i>EVConnectedPostSession</i> for EVSE.id = 1	
5. Execute Reusable State <i>EVDisconnected</i> for EVSE.id = 1	
6. Execute Reusable State <i>ParkingBayUnoccupied</i> for EVSE.id = 1	
Notes(s): The test system will wait the configured max timeout period to ensure the charging station does not reboot at this point.	
7. Execute Reusable State <i>StopAuthorized</i> for EVSE.id = 2	
Notes(s): If the TxStopPoint contains EnergyTransfer, PowerPathClosed, or Authorized: The test system will wait the <Configured Long Operation Time Out> to determine if the charging station resets because the transaction has ended.	
If the charging station resets proceed to step 11. Else proceed with step 8.	
8. Execute Reusable State <i>EVConnectedPostSession</i> for EVSE.id = 2	
Note(s): If TxStopPoint contains one of the following values; Authorized, EnergyTransfer, PowerPathClosed, DataSigned. Then the transaction will have ended at the EVConnectedPostSession state AND the Charging Station will proceed with resetting itself. Proceed to step 11. Else proceed with step 9.	
9. Execute Reusable State <i>EVDisconnected</i> for EVSE.id = 2	
Note(s): If TxStopPoint contains the value EVConnected. Then the transaction will have ended at the EVDisconnected state AND the Charging Station will proceed with resetting itself. Proceed to step 11. Else proceed with step 10	
10. Execute Reusable State <i>ParkingBayUnoccupied</i> for EVSE.id = 2	
Note(s): The transaction will end at this state, if it was not ended at an earlier state. Proceed to step 11.	
Notes(s): Step 10 will only be executed if TxStopPoint is ParkingBayOccupancy	
11. The Charging Station sends a BootNotificationRequest	12. The Test System responds with a BootNotificationResponse
13. The Charging Station notifies the CSMS about the current state of all connectors.	14. The Test System responds accordingly.

Tool validations
<p>* Step 2: Message ResetResponse - status <i>Scheduled</i></p> <p>* Step 11: Message BootNotificationRequest - reason <i>ScheduledReset</i></p> <p>* Step 13: Message: StatusNotificationRequest - If the transaction was stopped charging station reset at step 3 7, then – for the connector involved in the transaction: connectorStatus <i>Occupied</i> – for other connectors of this EVSE: connectorStatus <i>Available</i> or <i>Unavailable</i> - Else connectorStatus <i>Available</i> Message: NotifyEventRequest - eventData[0].trigger <i>Delta</i> - eventData[0].component.name <i>"Connector"</i> - eventData[0].variable.name <i>"AvailabilityState"</i> - If the transaction was stopped charging station reset at step 3 7, then – for the connector involved in the transaction: eventData[0].actualValue <i>Occupied</i> – for other connectors of this EVSE: eventData[0].actualValue <i>Available</i> or <i>Unavailable</i> - Else eventData[0].actualValue <i>"Available"</i></p>
Post scenario validations: - A message to report the state of a connector has been received for all connectors.

6.2.4. Page 107 - (2026-02) - TC_C_17_CS - Corrected wrong triggerReason [1113]

Wrong mention of triggerReason in step 5 has been changed to chargingState. Tool validation steps clarified.

Main (Test scenario)	
Charging Station	CSMS
1. The Test System closes the WebSocket connection AND does not accept a reconnect.	
<u>Manual Action:</u> Present valid idToken which is already configured in the Authorization Cache	
<u>Note(s):</u> The Test System will wait for 5 seconds	
2. The Test System accepts reconnection attempt from the Charging Station.	
<u>Note(s):</u> The Charging Station will empty its Transaction message queue. This will contain one or more TransactionEventRequest messages	
3. The Charging Station sends a TransactionEventRequest	4. The Test System responds with a TransactionEventResponse with idTokenInfo.status <i>Invalid</i> (if idToken is not omitted)
3. The Charging Stations empties its Transaction message queue: TransactionEventRequest	4. The Test System responds with a TransactionEventResponse
<u>Note(s) :</u> - This will contain one or more TransactionEventRequest messages	<u>Note(s) :</u> - The Test System will respond to the TransactionEventRequest containing the idToken, with idtokenInfo.status <i>Invalid</i>
5. The Charging Station sends a TransactionEventRequest with triggerReason <i>SuspendedEVSE</i> transactionInfo.chargingState <i>SuspendedEVSE</i>	6. The Test System responds with a TransactionEventResponse
<u>Note(s):</u> Steps 5 & 6 can occur before or during the sending of messages from the offline queue (steps 3 & 4).	

Tool validations * Step 5: Step 3: Message TransactionEventRequest A message with: All Message(s): TransactionEventRequest - offline must be true One of the Message(s): TransactionEventRequest - triggerReason Authorized - idToken.idToken <Configured valid_idtoken_idtoken> - idToken.type <Configured valid_idtoken_type> - offline True true * Step 5: A message with: - offline false or omitted - transactionInfo.chargingState SuspendedEVSE - triggerReason ChargingStateChanged

6.2.5. Page 124 - (2026-02) - TC_C_38_CS - Removed memory state IdTokenCached [1164]

There is no need to call memory state IdTokenCached if the Authorization Cache has been disabled. Thas has therefore been removed.

Test case name	Clear Authorization Data in Authorization Cache - Rejected
Test case Id	TC_C_38_CS
...	

Before (Preparations) Configuration State: AuthCacheEnabled is false (If implemented) Memory State: IdTokenCached for <Configured valid IdToken fields> Reusable State(s): N/a
--

[...]
NOTE: If the Charging Station supports ISO15118, this testcase needs to be executed using EIM.

6.2.6. Page 150/151 - (2026-02) - TC_C_26_CS - Clarification of main test and tool validation steps.

Main (Test scenario)	
Charging Station	CSMS
1. The Charging Station notifies the CSMS about the current state of all connectors.	2. The Test System responds accordingly.
3. The Charging Stations sends a TransactionEventRequest The Charging Station empties its Transaction message queue: TransactionEventRequest	4. The Test System responds with a TransactionEventResponse
Note(s): - The Charging Station will empty its Transaction message queue. This will contain one or more TransactionEventRequest messages	Note(s): - The Test System will respond to the TransactionEventRequest containing the idToken, with idtokenInfo.status Invalid

Main (Test scenario)	
5. The Charging Stations sends a TransactionEventRequest Note(s) : - This will contain chargingState SuspendedEVSE	6. The Test System responds with a TransactionEventResponse
Note(s) : Steps 5 & 6 can occur before or during the sending of messages from the offline queue (steps 3 & 4).	
5. 7. Execute Reusable State <i>StopAuthorized</i>	
6. 8. Execute Reusable State <i>EVConnectedPostSession</i>	
7. 9. Execute Reusable State <i>EVDisconnected</i>	
Tool validations	
* Step 1: Message: StatusNotificationRequest - connectorStatus must be <i>Occupied</i> Message: NotifyEventRequest - eventData[0].trigger must be <i>Delta</i> - eventData[0].actualValue must be <i>Occupied</i> - eventData[0].component.name must be <i>Connector</i> - eventData[0].variable.name must be <i>AvailabilityState</i> * Step 3: All Message(s): TransactionEventRequest - offline must be <i>true</i> One of the Message(s): TransactionEventRequest - chargingState must be <i>Charging</i> * Step 4: One of the Message(s): TransactionEventRequest - chargingState must be <i>SuspendedEVSE</i> * Step 5: TransactionEventRequest - chargingState must be <i>SuspendedEVSE</i>	
Post scenario validations: N/a	

6.2.7. Page 401 - (2026-02) - TC_L_07_CS - Path to non-existent firmware updated

Test case name	Secure Firmware Update - DownloadFailed
Test case Id	TC_L_07_CS
...	

Main (Test scenario)	
Charging Station	CSMS
2. The Charging Station responds with a UpdateFirmwareResponse	1. The Test System sends a UpdateFirmwareRequest with firmware.installDateTime <Current DateTime - 2 hours> firmware.location "does_not_exist_download" + <Configured firmware location> + "_does_not_exist" firmware.retrieveDateTime <Current DateTime - 2 hours> firmware.signingCertificate <Configured signingCertificate> firmware.signature <Configured signature> Note(s) : - The firmware location is mutated such that "does_not_exist_download" is pre-appended to the file path.
...	

6.2.8. Page 449 - (2026-02) - TC_M_24_CS - GetCertificateStatusRequest is not intended to be sent for the V2G leaf certificate

Updated note

	Text
Old	<u>Note(s)</u> : Step 1/2 are repeated for the V2G Charging Station (leaf), the subCA1 and subCA2 certificates.
New	<u>Note(s)</u> : - Step 1/2 are executed for both subCA1 and subCA2. - Step 1/2 are not intended to be executed for the V2G Charging Station (leaf), but it is not disallowed.

6.2.9. Page 552 - (2026-02) - TC_O_24_CS - Additions to tool validations

Test case name	Set Display Message - Second Alwaysfront priority
Test case Id	TC_O_24_CS
...	

Main (Test scenario)	
Charging Station	CSMS
2. The Charging Station responds with a SetDisplayMessageResponse	1. The Test System sends a SetDisplayMessageRequest with message.id <Generated displayMessageId> message.priority AlwaysFront
4. The Charging Station responds with a SetDisplayMessageResponse	3. The Test System sends a SetDisplayMessageRequest with message.id <Configured displayMessage2Id> message.priority AlwaysFront
6. The Charging Station responds with a GetDisplayMessagesResponse	5. The Test System sends a GetDisplayMessagesRequest with id <Configured displayMessageId>
6. 8. The Charging Station responds with a GetDisplayMessagesResponse	5. 7. The Test System sends a GetDisplayMessagesRequest with id <Configured displayMessage2Id>
7. 9. The Charging Station sends a NotifyDisplayMessagesRequest	8. 10. The Test System responds with a NotifyDisplayMessagesResponse .
<u>Note(s)</u> : - If tbc is True at Step 7 9 then step 7 9 and 8 10 will be repeated - The message from step 1 is NOT displayed anymore and is replaced by the message from step 5 7 .	

Tool validations
* Step 2: Message SetDisplayMessageResponse - status Accepted * Step 4: Message SetDisplayMessageResponse - status Accepted * Step 6: Message GetDisplayMessagesResponse - status Unknown * Step 6: Step 8: Message GetDisplayMessagesResponse - status Accepted * Step 7: Step 9: Message NotifyDisplayMessagesRequest - requestId <Generated requestId>

Tool validations
Post scenario validations: - N/a

749

6.2.10. Page 585 - (2026-02) - Reusable state Booted now accepts StatusNotification Unavailable [1162]

Booted

State	Booted
System under test	Charging Station
Description	This state will reset or power cycle the Charging Station, depending on the testcase. The charging station ends in a state where it is booted back up and is in idle mode.

Before (Preparations)
Configuration State: N/a
Memory State: N/a
Reusable State(s): N/a

Main (Test scenario)	
Charging Station	CSMS
Manual Action: Power cycle the Charging Station. OR execute step 1 and 2, depending on the testcase.	
2. The Charging Station responds with a ResetResponse with status Accepted	1. The Test System sends a ResetRequest
3. The Charging Station notifies the CSMS about the unavailability of all connectors. <u>Note:</u> This step is optional.	4. The Test System responds accordingly.
5. The Charging Station sends a BootNotificationRequest	6. The Test System responds with a BootNotificationResponse with status Accepted
7. The Charging Station notifies the CSMS about the current state of all connectors.	8. The Test System responds accordingly.
9 The Charging Station sends a SecurityEventNotificationRequest	10 The Test System responds with a SecurityEventNotificationResponse

Tool validations
<p>* Step 2:</p> <p>Message: ResetResponse</p> <ul style="list-style-type: none"> - status <i>Accepted</i> <p>* Step 3:</p> <p>Message: StatusNotificationRequest</p> <ul style="list-style-type: none"> - connectorStatus <i>Unavailable</i> - evseld not <i>0</i> - connectorId not <i>0</i> <p>Message: NotifyEventRequest</p> <ul style="list-style-type: none"> - eventData[0].trigger <i>Delta</i> - eventData[0].actualValue <i>"Unavailable"</i> - eventData[0].component.name <i>"Connector"</i> - eventData[0].variable.name <i>"AvailabilityState"</i> <p>* Step 7:</p> <p>Message: StatusNotificationRequest</p> <ul style="list-style-type: none"> - connectorStatus <i>Available</i> - evseld not <i>0</i> - connectorId not <i>0</i> <p>Message: NotifyEventRequest</p> <ul style="list-style-type: none"> - eventData[0].trigger <i>Delta</i> - eventData[0].actualValue <i>"Available"</i> - eventData[0].component.name <i>"Connector"</i> - eventData[0].variable.name <i>"AvailabilityState"</i> <p>* Step 9:</p> <p>Message: SecurityEventNotificationRequest</p> <ul style="list-style-type: none"> - type must be <i>StartupOfTheDevice</i> OR <i>ResetOrReboot</i>
<p>Post scenario validations:</p> <p>State is <i>Booted</i></p>

6.3. CSMS

Currently no errata for Part 6 CSMS test cases.