TERRA AC WALLBOX

OCPP 1.6

Implementation Overview

Contents

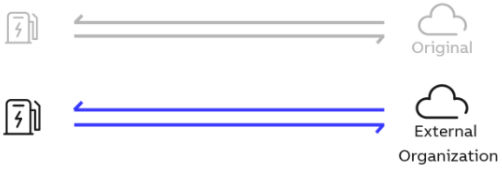
1. Overview

ABB Terra AC chargers support OCPP 1.6 J. This document describes OCPP 1.6 functionality supported by ABB Terra AC chargers according to OCPP protocol specification.

ABB has implemented OCPP 1.6-J version, which means using JSON over WebSocket. The charger is connected to the ABB ChargerSync server by default. The charger is either connected to a 3rd party/external OCPP Server or to ChargerSync server. The connection to the ChargerSync portal allows for efficient remote support and enables additional features next to OCPP. This concept is referred to as below.

Link to the ABB ChargerSync server:

Link to a 3rd party OCPP backend server:



2. OCPP server configuration

Contact your local ABB sales representative to arrange access to TerraConfig App account and facilitate company creation and new OCPP configuration via the ChargerSync Portal.

For more information refer section 4.

2.1. ABB ChargerSync server

By default, charger is connected to ABB ChargerSync server. And this ChargerSync server can be accessed by ChargerSync mobile application as well as web portal.

2.2. OCPP external server configuration

Once you have access for TerraConfig App, the below steps need to be performed for each charger commissioning that needs connection to a 3rd party/external OCPP backend.

1. Download TerraConfig app and use the credentials sent in the email, generated by the creation of the account.
2. Pair the TerraConfig app with the charger and check firmware version. It's always recommended that charger has latest firmware version.
3. Make sure the charger is connected to the internet (over Wi-Fi, LAN or 4G).
4. Enable external OCPP server and choose the correct backend URL which has been preconfigured in the portal. Then press configure and afterwards OK.
5. Check via Device info that the URL is the correct one and that connection has been established. Via OCPP logs (of the 3rd party backend) validate that BootNotification is successfully sent.
6. Run some remote commands to confirm good communication between charger and backend.

2.3. WebSocket communication

For the connection between a Charge Point(charger) and a Central System (OCPP Server) using OCPP-J, the Central System acts as a WebSocket server and the Charge Point acts as a WebSocket client.

2.3.1. Client request

The following is an example of an opening HTTP request of an OCPP-J connection handshake:

GET */webServices/ocpp/CP3211 HTTP/1.1*

Host: *some.server.com:33033*

Upgrade: websocket

Connection: Upgrade

Sec-WebSocket-Key: *x3JJHMbDL1EzLkh9GBhXDw==*

Authorization: Basic <Base64 encoded(<ChargePointId>:<AuthorizationKey>)>*

Sec-WebSocket-Protocol: *ocpp1.6, ocpp1.5*

Sec-WebSocket-Version: 13

The bold parts are found as such in every WebSocket handshake request, the other parts are specific to this example. In this example, the Central System's OCPP-J endpoint URL is "ws://some.server.com:33033/webServices/ocpp". The Charge Point's unique identifier is "CP3211", so the path to request becomes "webServices/ocpp/CP3211".

Remark:

*The *AuthorizationKey (i.e.) password* was configured in TerraConfig portal and sent to charger via Bluetooth using TerraConfig App.

2.3.2. Server response

Upon receiving the Charge Point's request, the Central System has to finish the handshake with a response as described in RFC6455.

So, if the Central System accepts the above example request and agrees to using OCPP 1.6J with the Charge Point, the Central System's response will look as follows:

HTTP/1.1 101 Switching Protocols

Upgrade: websocket

Connection: Upgrade

Sec-WebSocket-Accept: *s3pPLMBiTxaQ9kYGzzhZRbK+xOo=*

Sec-WebSocket-Protocol: *ocpp1.6*

The bold parts are found as such in every WebSocket handshake response, the other parts are specific to this example.

2.3.3. WebSocket ping in relation to OCPP Heartbeat

The WebSocket specification defines Ping and Pong frames that are used to check if the remote endpoint is still responsive. In practice this mechanism is also used to prevent the network operator from quietly closing the underlying network connection after a certain period of inactivity. This WebSocket feature can be used as a substitute for most of the OCPP Heartbeat messages but cannot replace all of its functionality.

Remark:

Charger supports sending PING every certain time interval. The interval range is 0, 10 to 65535 seconds, 0 disables the PING, default value is 60 seconds.

Charger will also respond with PONG to the PING request from the server.

3. Supported Feature profiles

The implementation is following OCPP 1.6 specification of Open Charge Alliance. According to OCPP 1.6 specification all of features and associated messages are grouped into Feature Profiles.

OCPP 1.6 specified following Feature profiles:

Profile name	Description	Implementation Mandatory
Core	Basic Charge Point functionality comparable with OCPP1.5 without support for firmware updates, local authorization list management and reservations.	Required
Firmware Management	Support for firmware update management and diagnostic log file download.	Optional
Local Auth List Management	Features to manage the local authorization list in Charge Points.	Optional
Reservation	Support for reservation of a Charge Point	Optional
Smart Charging	Support for basic Smart Charging	Optional
Remote Trigger	Support for remote triggering of Charge Point initiated messages	Optional

3.1. Messages

Please see below which messages are supported per OCPP feature profile.

Message	Supported (Y/N)	Comment
Core profile		
Authorize	Y	Example when two different online cards: card #1 swiped and starts, card #2 swipes, card #2 is checked by back-end card, if accepted charging stops. If rejected charging continues.
BootNotification	Y	
ChangeAvailability	Y	
ChangeConfiguration	Y	
ClearCache	Y	The cache is empty, while the charger received the command, it will response accept but do nothing.
DataTransfer	Y	While the charger connects to customer's own back end, customer could use this message for log transfer. NOTE: Please refer OCPP Data Transfer document. This option may become obsolete moving forward for log transfer as GetDiagnostics message is supported.
GetConfiguration	Y	Before FW 1.3 the charger will only response with support keys. After FW1.3 charger will response with all keys, in case the key not supported, the charger will response "unknown" Refer section Supported keys.
HeartBeat	Y	
MeterValues	Y	ABB supports following Measurand types for AC: • Energy.Active.Import.Register • Current.Import • Voltage • Power.Active.Import • Current.Offered
RemoteStartTransaction	Y	
RemoteStopTransaction	Y	
Reset	Y	Chargers support hard reset and soft reset. Hard reset fully reboot charger. The resets gracefully stop charging session if one is in progress before resetting. Soft reset gracefully stops charging session if one is in progress before resetting. Then the charger gracefully disconnects from the server. After disconnection, the charger will reboot itself.
StartTransaction	Y	
StatusNotification	Y	
StopTransaction	Y	
UnlockConnector	Y	Message is supported only to socket variants, upon receiving this message, socket variants

Message	Supported (Y/N)	Comment
		charger will release the E-lock of socket. If send the message to cable variants, the charger will response with NotSupported.
Firmware Management profile		
GetDiagnostics	Y	Supports 7-day logs with 300 lines/less than 25kb per day. Charger uploads the files by HTTP/HTTPS based on the server connection.
DiagnosticsStatusNotification	Y	Support status: Uploading, Uploaded, Upload-Failed, Idle.
FirmwareStatusNotification	Y	Charger will response the status: • Downloading • Installed • DownloadFailed • InstallationFailed
UpdateFirmware	Y	
Local Auth List Management		
GetLocalListVersion	Y	
SendLocalList	Y	Each list is limited to 8 ID tag, each ID tag with max 20 characters; The charger has a total limit of 16 ID tags.
Reservation*		
CancelReservation	N	Planned in roadmap.
ReserveNow	N	Planned in roadmap.
Smart charging*		
ClearChargingProfile	Y	
GetCompositeSchedule	Y	
SetChargingProfile	Y	ChargeProfileMaxStackLevel = 16 for display models. ChargeProfileMaxStackLevel = 3 for non-display models.
Remote Trigger*		
TriggerMessage	Y	Chargers supports below MessageTrigger: • BootNotification • DiagnosticsStatusNotification • FirmwareStatusNotification • Heartbeat • MeterValues • StatusNotification
Improved Security*		
SecurityEventNotification	Y	Chargers supports below SecurityEventNotification: • FirmwareUpdated • SettingSystemTime • StartupOfTheDevice • ResetOrReboot • MemoryExhaustion • SecurityLogWasCleared

Message	Supported (Y/N)	Comment
		• InvalidFirmwareSignature • AttemptedReplayAttacks
GetLog	Y	Chargers only supports request SecurityLog, DiagnosticsLog to be supported
LogStatusNotification	Y	Charger will response the status: • Uploading • Uploaded • UploadFailure

*Note: Section “3.3. Feature Profiles” of OCPP specification [1] contains the mapping of message versus feature profiles. And it seems that there is some error on that mapping in the OCPP specification. The above table is aligned with “Test case document OCTT for OCPP 1.6” from OCA.

3.2. Standard Configuration keys

3.2.1. Supported keys

Key Name	Required/ Optional	Description	Type	Accessibility	Default Value
Core profile					
ClockAlignedDataInterval	required	Size (in seconds) of the clock-aligned data interval. This is the size (in seconds) of the set of evenly spaced aggregation intervals per day, starting at 00:00:00 (midnight). For example, a value of 900 (15 minutes) indicates that every day should be broken into 96 15-minute intervals. The range of this value: 0, 30 – (86400-1)	Integer	RW	0
ConnectionTimeOut	required	Interval(from successful authorization) until incipient charging session is automatically canceled due to failure of EV user to (correctly) insert the charging cable connector(s) into the appropriate connector(s). The range of this value: 10 - 240	Integer	RW	120
GetConfigurationMaxKeys	required	The number of configuration keys requested in a single PDU may be limited by the Charge Point. This maximum can be retrieved by reading this configuration key.	Integer	R	20
HeartbeatInterval	required	Interval of inactivity (no OCPP exchanges) with central system after which the Charge Point should send a Heartbeat.req PDU. If the interval less than 10, the AC charger will accept but execute 10	Integer	RW	120
MeterValuesAlignedData	required	Clock-aligned measurand(s) to be included in a MeterValues.req PDU, every ClockAlignedDa- taInterval seconds. Supported value: Current.Import, Current.Offered, Energy.Active.Import.Register, Energy.Active.Import.Interval, Power.Active.Import, Power.Offered, Voltage.	CSL	RW	Energy.Active.Import.Register

Key Name	Required/ Optional	Description	Type	Accessibility	Default Value
MeterValuesAligned-DataMaxLength	required	Maximum number of items in a MeterValuesAlignedData Configuration Key.	Integer	R	4
MeterValuesSampleInterval	required	Interval between sampling of metering (or other) data, intended to be transmitted by "MeterValues" PDUs. The range of this value: 0, 4 - 65534 If the interval less than 4, the AC charger will reject.	Integer	RW	30
LocalAuthorizeOffline	required	Controls whether a Charge Point will authorize a user when offline using the Authorization Cache and/or the Local Authorization List.	boolean	RW	TRUE
LocalPreAuthorize	required	Controls whether a Charge Point will use the Authorization Cache and/or the Local Authorization List to start a transaction without waiting for an authorization response from the Central System.	boolean	RW	FALSE
NumberOfConnectors	required	The number of physical charging connectors of this Charge Point.	Integer	R	1
SupportedFeatureProfiles	required	A list of supported Feature Profiles. Possible profile identifiers: Core, FirmwareManagement, LocalAuthListManagement, Reservation, SmartCharging and RemoteTrigger.	CSL	R	Core, FirmwareManagement, LocalAuthicationListManagement, SmartCharging, RemoteTrigger
WebSocketPingInterval	optional	Only relevant for websocket implementations. 0 disables client side websocket Ping/Pong. In this case there is either no ping/pong or the server initiates the ping and client responds with Pong. Positive values are interpreted as number of seconds between pings. Negative values are not allowed. ChangeConfiguration is expected to return a REJECTED result. The range of this value: 0, 10 – 65535	Integer	RW	60

Key Name	Required/ Optional	Description	Type	Accessibility	Default Value
AllowOfflineTxForUnknownId	optional	When offline, a Charge Point may allow automatic authorization of any "unknown" identifiers that cannot be explicitly authorized by Local Authorization List or Authorization Cache entries. Identifiers with status other than "Accepted" (Invalid, Blocked, Expired) must be rejected. Now the charger will not allow any ID except in local authentication list while it is offline	boolean	RW	False
AuthorizeRemoteTxRequests	required	Whether a remote request to start a transaction in the form of a RemoteStartTransaction.req message should be authorized beforehand like a local action to start a transaction. Now the charger will not send the authorize.req	boolean	RW	False
Local Authorization List Management					
LocalAuthListEnabled	required	Whether the Local Authorization List is enabled	boolean	RW	TRUE
LocalAuthListMaxLength	required	Maximum number of identifications that can be stored in the Local Authorization List	Integer	R	16
SendLocalListMaxLength	required	Maximum number of identifications that can be send in a single SendLocalList.req	Integer	R	8
Smart charging profile					
ChargeProfileMaxStackLevel	required	Max StackLevel of a Charging. The number defined also indicates the max allowed number of installed charging schedules per Charging Purposes.	Integer	R	Based on the charger model
ChargingScheduleAllowedChargingRateUnit	required	A list of supported quantities for use in a ChargingSchedule. Allowed values: 'Current' and 'Power'.	CSL	R	Current,Power
ChargingScheduleMaxPeriods	required	Maximum number of periods that may be defined per ChargingSchedule.	Integer	R	25
MaxChargingProfilesInstalled	required	Maximum number of Charging profiles installed at a time.	Integer	R	Same as ChargeProfileMaxStackLevel
ResetRetries	required	Number of times to retry an unsuccessful reset of the Charge Point. Charger now only supports value 0.	Integer	RW	0
TransactionMessageAttempts	required	How often the Charge Point should try to submit a transaction-related message when the Central System fails to process it.	Integer	RW	10

Key Name	Required/ Optional	Description	Type	Accessibility	Default Value
		Now the transaction data will always attempt to send to central system until it response			
TransactionMessageRetryInterval	required	How long the Charge Point should wait before re-submitting a transaction related message that the Central System failed to process.	Integer	RW	10S
MaxEnergyOnInvalidId	optional	Maximum energy in Wh delivered when an identifier is invalidated by the Central System after start of a transaction.	integer	RW	1 Kwh
StopTransactionOnInvalidId	required	Whether the Charge Point will stop an ongoing transaction when it receives a non-Accepted authorization status in a StartTransaction.conf for this transaction. Now the default value is true.	Boolean	RW	TRUE
MeterValuesSampledData	required	Sampled measurands to be included in a MeterValues.req PDU, every MeterValueSampleInterval seconds. Supported value: Current.Import, Current.Offered, Energy.Active.Import.Register, Power.Active.Import, Voltage.	CSL	RW	Energy.Active.Import.Register, current import power active import, current.Offered Voltage
MeterValueSampleInterval	required	Interval between sampling of metering (or other) data, intended to be transmitted by "MeterValues" PDUs. For charging session data (ConnectorId>0), samples are acquired and transmitted periodically at this interval from the start of the charging transaction. A value of "0" (numeric zero), by convention, is to be interpreted to mean that no sampled data should be transmitted.	Integer	RW	30
MeterVaues-SampledDataMaxLength	Optional	Maximum number of items in a MeterValuesSampledData Configuration Key.	Integer	R	5

3.2.2. To be supported keys

Key Name	Required/ Optional	Description	Type	Accessibility	Default Value
Core profile					
ConnectorPhaseRotation	required	For individual connector phase rotation infor- mation, the Central System may query the Con- nectorPhaseRotation configuration key on the Charging Point via GetConfigura- tion. The Charge Point shall report the phase rotation in re- spect to the grid connection.	CSL	RW	Unknown
StopTxnAlignedData	required	Clock-aligned periodic measurand(s) to be included in the TransactionData element of Stop- Transaction.req Meter- Values.req PDU for every ClockAlignedDataInterval of the charging session.	CSL	RW	Unknown
StopTxnSampledData	required	Sampled measurands to be included in the Trans- ac- tionData element of StopTransaction.req PDU, every Meter- ValueSampleInterval seconds from the start of the charging session	CSL	RW	Unknown
StopTxnSampledDataMaxLength	optional	Maximum number of items in a StopTxnSampledData Con- figuration Key.	Integer	R	Unknown
StopTransactionOnEVSideDiscon- nect	required	When set to true, the Charge Point shall administratively stop the transaction when the cable is unplugged from the EV. NOTE: this parameter is not being used, Transaction will al- ways stop on EV disconnect or even before.	boolean	RW	Unknown
UnlockConnectorOnEVSideDis- connect	required	When set to true, the Charge Point shall unlock the cable on Charge Point side when the cable is unplugged at the EV. NOTE: not applicable for ABB TerraAC chargers, not imple- mented	boolean	RW	Unknown

3.2.3. Not supported keys

Key Name	Required/ Optional	Description	Type
Core profile			
AuthorizationCacheEnabled	optional	A Charge Point may implement an Authorization Cache that autonomously maintains a record of previously presented identifiers that have been successfully authorized by the Central System.	boolean
MinimumStatusDuration	optional	The minimum duration that a Charge Point or Connector status is stable before a StatusNotification.req PDU is sent to the Central System.	integer
BlinkRepeat	optional	Number of times to blink Charge Point lighting when signaling	integer
ConnectorPhaseRotationMaxLength	optional	Maximum number of items in a ConnectorPhaseRotation Configuration Key	integer
LightIntensity	optional	Percentage of maximum intensity at which to illuminate Charge Point lighting	integer
StopTxnAlignedDataMaxLength	optional	Maximum number of items in a StopTxnAlignedData Configuration Key.	integer
SupportedFeatureProfilesMaxLength	optional	Maximum number of items in a SupportedFeatureProfiles Configuration Key.	integer
ConnectorSwitch3to1PhaseSupported	optional	If defined and true, this Charge Point support switching from 3 to 1 phase during a charging session.	boolean
Reservation profile			
ReserveConnectorZeroSupported	optional	If this configuration key is present and set to true: Charge Point support reservations on connector 0.	boolean

3.3. Custom Configuration keys

Key Name	Description	Type	Accessibility	Default Value
FreevendEnabled	In Free Vend mode authorization is disabled and charging could be started without authorization. When this mode is enabled, Authorize message will not be sent to Central System. StartTransaction message will be sent as usually at the beginning of charging session.	boolean	RW	TRUE
FreevendIdTag	In Free Vend mode, use this key to set idTag. Max length is 20 like IdToken type. Central system should be configured to accept StartTransaction with idTag configured for Free Vend mode.	String	RW	Serial number
TimeOffset	This is used for display/local time purposes only. Configured current local time offset in the UTC time offset format. Let say for US, 5 hours ahead of UTC time. So, the value to set "-05:00". If this is not set/empty, it will show in UTC format.	String	RW	“-05:00”
NextTimeOffsetTransitionDateTime	This is used for display/local time purposes only. On this date time, the clock displayed to the EV driver will be given the new offset as configured via 'TimeOffsetNextTransition'. For example, "2022-03-28T02:00:00+01:00". It is represented according to the ISO8601 standard.	String	RW	“”
TimeOffsetNextTransition	This is used for display/local time purposes only. New offset that will be set on the next time offset transition as configured via 'NextTimeOffsetTransitionDateTime'. For example, "-04:00"	String	RW	“”

3.4. Security

3.4.1. Encryption

In addition to network level security, ABB OCPP 1.6 implementation supports OCPP-J over TLS security. TLS 1.2 is supported. It is up to Central System operator to decide if TLS with Web-Socket (WSS) is used or not. No additional configuration changes are required to enable it. For more information on encryption with OCPP 1.6-J please see chapter “6.2.1 Encryption” of [2].

3.4.2. Authentication

ABB OCPP 1.6 implementation supports basic HTTP authentication. Username equals charge point ID and password/authorization keys can optionally be set during installation.

Charger supports write-only OCPP AuthorizationKey which is maximum of 20 bytes where central system could change the key using ChangeConfiguration.req.

For more information on OCPP 1.6-J authentication please see chapter “6.2.2 Authentication” of [2].

3.5. Certification maturity

This section provides a summary towards certification readiness of Terra AC charger.

Functionality	Full Certificate	Subset Certificate	TAC support maturity
Core	Mandatory	Mandatory	Complete
Firmware Management	Mandatory	Optional	Complete
Smart Charging	Mandatory	Optional	Complete
Reservation	Mandatory	Optional	Planning
Local Auth List Management	Mandatory	Optional	Complete
Remote Trigger	Mandatory	Optional	Complete

Remark:

Refer “Messages” section for more details.

4. Additional Information

4.1. Listing of related documents

Ref #	Document Kind, Title	Document No.
1	Open Charge Point Protocol 1.6 edition 2 FINAL, 2017-09-28	
2	Open Charge Point Protocol JSON 1.6, OCPP-J 1.6 Specification	
3	Improved security for OCPP 1.6-J edition 2 FINAL, 2020-03-31	
4	TerraConfig App & Portal	ABB internal

Ref #	Document Kind, Title	Document No.
5	Accounts and 3 party OCPP backend configuration	ABB internal
6	OCPP Data Transfer document	Request your LSU
7	The Electric Vehicles (Smart Charge Points) Regulations 2021 (legislation.gov.uk)	

5. Addendum

5.1. Firmware upgrade from external OCPP server

ABB LSU will provide bin files of a firmware release from ABB sales SharePoint link to the customer. And customer need to use “UpdateFirmware” message with the location field where firmware files are stored. Please refer OCPP1.6 specification [1] section “6.55. UpdateFirmware.req”. In order to start a process of download, OCPP server should send command to the charger to trigger the process.

OCPP command example:

<MessageTypeId>, "<UniqueId>", "<Action>", {<Payload>}]

[2,"256530071089713152","UpdateFirmware",{"location":"https://down.XXX.com/CE-N-22-1.4.2/?524&TAC22W910012345678","retrieveDate":"2021-12-15T13:18:55.000Z"}]

Note:

The location of bin files should be same as below:

https://Domain name+ port number + Relative path address of the update bin files+/?+ quantity of bin files&package code

Only this format can be identified by charger. Port number is unnecessary.

URL Example:

https://down.XXX.com/CE-N-22-1.4.2/?524&TAC22W910012345678

- CE-N-22-1.4.2 is a folder on the server for bin files.
- The number of 524 is the quantity of slices of the updating package.
- TAC22W910012345678 is FW package code which is an optional parameter.
- The protocol should same as server connection type such as http or https. And charger will initiate GET request to fetch files one by one.

5.2. Regional specific interface support

5.2.1. UK - The Electric Vehicles (Smart Charge Points) Regulations 2021

Charger provides below OCPP interface options to adhere compliance with regulation. This information facilitates customer/end user if they prefer enable/control regulation related actions. Since this is provided as an additional option, its customer/end user responsibility to configure the chargers according to their preferences. For more details about regulation refer The Electric Vehicles (Smart Charge Points) Regulations 2021 (legislation.gov.uk).

5.2.1.1. Randomised delay - Custom configuration keys

Key Name	Description	Type	Accessibility	Default Value
RandomDelay	RandomDelay has three parameters: delay duration, random mode and enable mode. OCPP send message sequence according to {Delay duration, Random mode, Enable mode} **Delay duration**: The range is from 0 to 1800 in seconds. **Random mode**: Below mode of operation possible. • 1 - delay duration is always fixed. • 2 - delay duration is randomly generated (with in the Delay duration) by charger for each transaction. **Enable mode**: Below mode of operation possible. • 0 - Disable random delay. • 1 - Enable random delay and it will take effect for each start of transactions and the start of each schedule (peak/off-peak hour). • 2 - Enable random delay and it will take effect only for the start of each schedule (peak/off-peak hour). Remark: Similar setting is influenced by mobile app where Random Mode to 2 and Enable mode to 1 set implicitly.	CSL	RW	Delay duration :600 Random Mode:2 Enable Mode: 1
RandomDelayCancel	If set to true, RandomDelay will be cancelled and starts charging directly. After each transaction is finished, this key will be set to default value automatically. Remarks: This provides similar behavior provided by Charger Sync mobile app.	boolean	RW	False

5.2.1.2. Off-peak charging - Example preset charging schedule

Regulation indicates that ”peak hours” means 8am to 11am on weekdays and 4pm to 10pm on weekdays. And here is an example OCPP command (one of the ways) how charger could be configured to incorporate pre-set default charging hours which are outside of peak hours.

ChargingProfile			
chargingProfileId	14		
stackLevel	14		
chargingProfilePurpose	TxDefaultProfile		
chargingProfileKind	Recurring		
recurrencyKind	Daily		
chargingSchedule			
	duration	86400	
	startSchedule	2022-06-15T00:00:00.0Z	
	chargingRateUnit	A	

	chargingSchedu-lePeriod		
		startPeriod	0
		limit	32
		startPeriod	28800
		limit	0
		startPeriod	39600
		limit	32
		startPeriod	57600
		limit	0
		startPeriod	79200
		limit	32
chargingProfileId	15		
stackLevel	15		
chargingProfilePur-pose	TxDefaultProfile		
chargingProfileKind	Recurring		
recurrencyKind	Weekly		
chargingSchedule			
	duration	172800	
	startSchedule	2022-06-18T00:00:00.0Z	
	chargingRateUnit	A	
	chargingSchedu-lePeriod		
		startPeriod	0
		limit	32

Remarks:
Example is provided with following assumptions.

- stackLevel number intended to indicate hierarchy stack of profiles where higher values have precedence over lower values.
- startSchedule intended to start from 15-July-2022 and stackLevel 15 is to support exception on weekend days.

5.2.2. United States (US) specific support

5.2.2.1. Touch screen model - Custom configuration keys

Key Name	Description	Type	Accessibility	Default Value
ScreenDisplaySetting	It has two values of integer type. **ScreenOffTime**: Select how long you want your charger display to wait before turning the screen off. The possible value is 10 to 300 seconds. When value is 0, then display will be always ON. If the value is other than possible range defined, it will be 'Rejected'. **Brightness**: possible range to adjust screen brightness is 10 to 100. If the value is other than possible range defined, it will be 'Rejected'.	CSL	RW	120, 100
ScreenBrightnessLevel	This key change the screen brightness. The value of key is 1 or 2, other values will be rejected. Level 1 mean the brightness is 100%, level 2 mean the brightness is 50%.	Integer	RW	1

5.3. Security digital certificate

This section provides complementary information for the Terra AC charger (TAC) user to understand brief information about security/digital certificate to establish secured communication with their own OCPP server/backend/Central Systems via OCPP communication protocol. And, whether security needed or not, and different OCPP security profiles are out the scope.

1. What is certificate?

   Simple analogy is Passport or University degree certificate. A well know **trusted** bodies/authorities provide **identity of the owner** which is **publicly** accepted.
   Now, with the security context, certificate is used as **identity** of individual, computer, and other entities on the network and **public key**.
   This certificate is intended for the following purposes:
   - Proves your identity to a remote computer
   - Ensures the identity of a remote computer

   And a certificate contains below general information while keep aside the detailed technical information.
   Issued to :
   Issue from:
   Valid from x to y date.

2. Why certificate needed?

   To establish a secured connection, a system uses Secure Socket Layer (SSL) also known as Transfer Layer Security (TLS) protocol communication which is based on public key cryptography. In public key cryptography, a matching pair of keys is used; one for encryption and the other for decryption. One of the keys is called the **public key** (can be sent over the network to another system). The other is called the private key (kept secretly by the owner).

   The purpose of digital certificate is to ascertain the public key belongs to someone who claim who own. The certificate contains the name of the person (or the organization), together with the public key.

3. Why is certificate needed for TAC?

   OCPP specification provides a way to achieve security measures for charge point (TAC) and Central System (let say OCPP sever), to establish a secure connection between charger and OCPP server, also to validate/authenticate whether charger is communicating to correct OCPP sever.

   The secure/encrypted communication shall be achieved by web socket secure(wss), whereas OCPP server authentication is based on TLS and public key cryptography using X.509 certificates.

4. Who issues certificate?

   Certificate Authority (CA)/Third Party:
   CA is a trusted organization who validate the identity of the holder and provide a certificate. For example, to name few Symantec, GoDaddy, DigiCert etc., who sign and provide a digital certificate based on applicant/user Certificate Signing Request (CSR).

   Self-signed certificate:
   This is created, issue, signed by themselves. A self-signed certificate can be used in private network.

   A Certificate Authority (CA) signs (i.e., encrypts) the certificate using its private key. Consequently, the certificate can only be decrypted using CA's public key. CAs are considered trustworthy, and their public key are pre-installed into the browser.
   One single CA (or a few CAs) to sign all the certificates and CAs are organized in hierarchy. The root CA, whose public key is pre-installed inside the browser, signs the certificate of sub-CAs. The sub-CAs can sign the certificate of sub-sub-CAs. The sub-sub-CAs can then sign the certificate of end-users. Because of this hierarchical structure, the certificate verification process involves a chain of certificates, all the way back to a root CA.

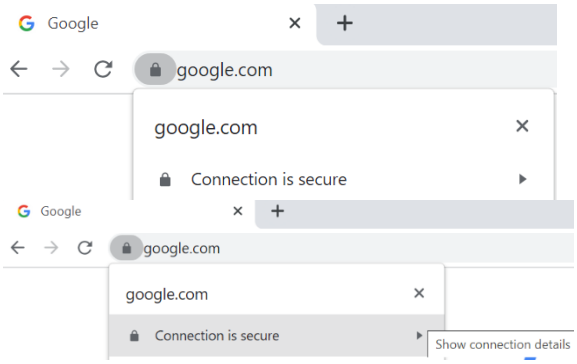
   For TAC, its recommended to use CA signed certificate however it’s not a mandate. And root CA is used for authentication.

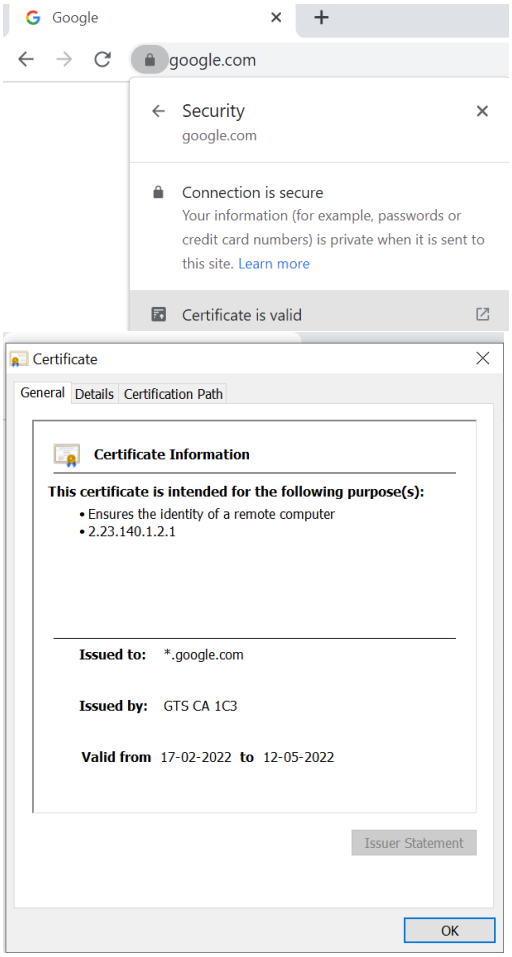
5. How to download my server root certificate?

   During installation process, installer would request to provide your OCPP server URL and its certificate.

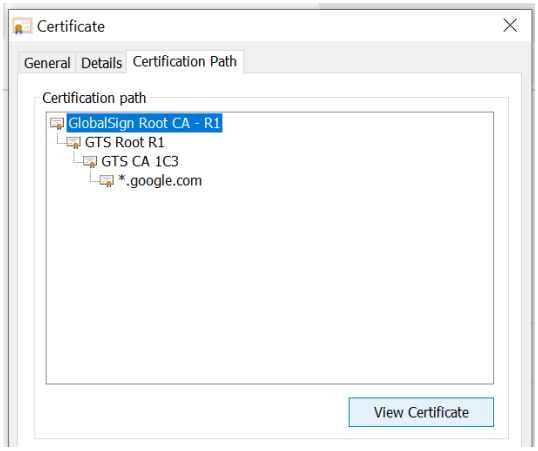
   Let us consider, google.com is your server, below steps would help to get the certificate of your server in chrome browser. You will be able to relate if you are using different browser.

   a. Click the secure(lock) symbol in the server URL and navigate further to open certificate as shown in the snap.

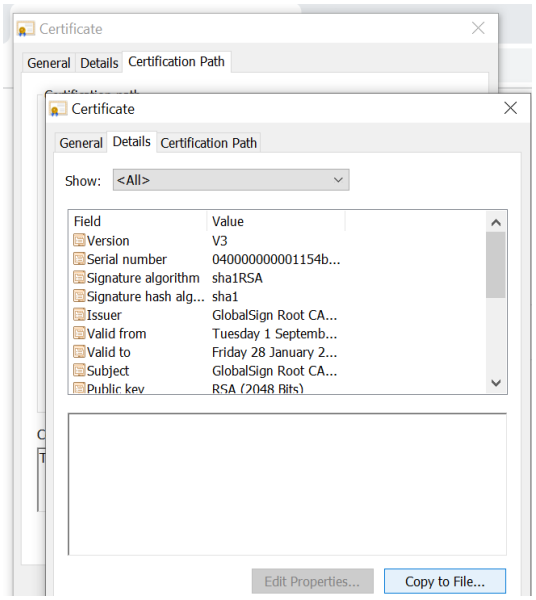




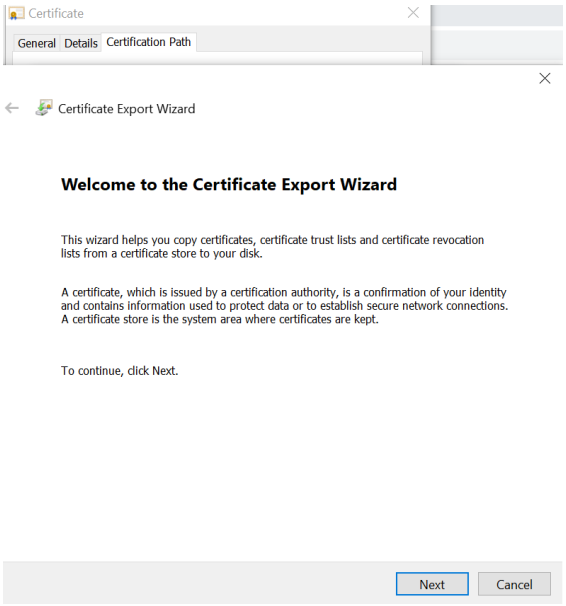
b. Go to certificate path tab and select the Root CA. And then click view certificate.



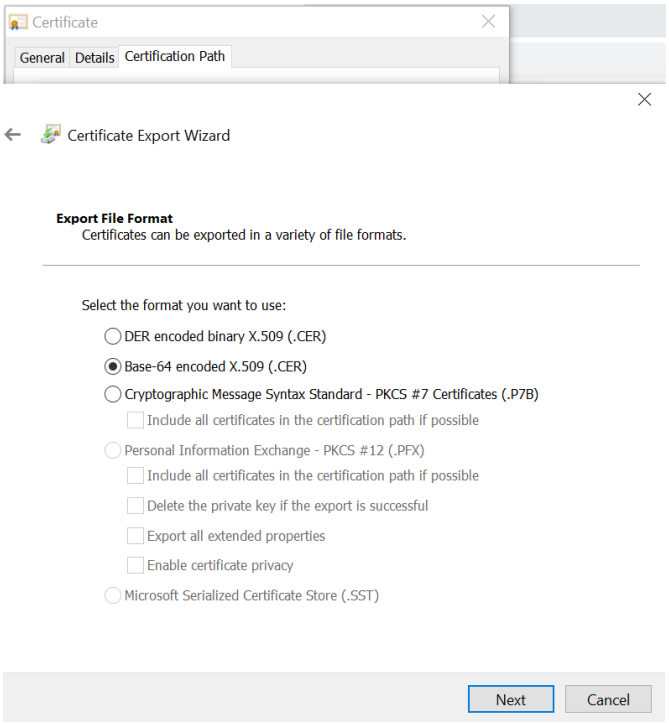
c. Go to the “Details” tab and click copy to file.



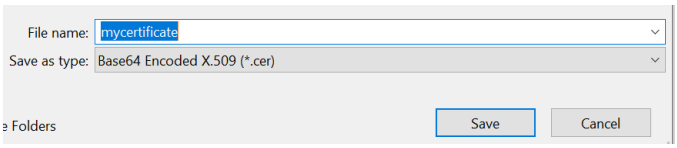
d. Click Next and continue



e. Select Base 64 encoded as shown, proceed to Next.



f. Save the certificate file as below (i.e.) .cer extension.



g. Finish the wizard. Now you have certificate of your sever to share with installer to configure with TAC charger.

6. Revisions

Rev.	Page (P) Chapt. (C)	Description	Date Dept./Init.
1.6.1		No technical changes from 1.6 document version(FW1.5.2). Only document restructuring. And section 5 is added for informative.	03/17 PM VJ
1.7		Firmware 1.6.3 updates. Refer release note for more details. And added a new section to capture regional specific details.	July 2022 PM/VJ
1.8		Firmware 1.6.6 updates. FreeVendEnabled, FreeVendIdTag change to FreevendEnabled, FreevendIdTag Support SecurityEventNotification, GetLog, LogStatusNotification	May 2023 PM JV