1. Configuration Key Names & Values

1.1. FreeChargingIdTag

Required/optional	required
Accessibility	RW
Type	CiString20Type
Description	idToken used as part of StartTransaction.req when no other idToken is specified. This SHOULD only be used by the Charge Point when FreeCharging is enabled.

1.2. FreeCharging

Required/optional	required
Accessibility	RW
Type	boolean
Description	Whether the Charge Point will start charging without requiering idTag authorization.

1.3. LockCablePermanently

Required/optional	required
Accessibility	RW
Type	boolean
Description	Whether the Charge Point SHALL lock the cable in place after insertion, and thereafter act as if the cable is permanently attached. If true then the UnlockConnectorOnEVSideDisconnect configuration key SHALL be false and read only.

1.4. PowerActiveImportDeltaForEagerMeterValueSample

Required/optional	required
Accessibility	RW
Type	integer
Unit	W
Description	Maximum change in Power.Active.Import since last MeterValue related to MeterValuesSampledData before a new Sample.Periodic is sent and MeterValueSampleInterval is reset. A value of 0 disables this behaviour. For example, if set to 200 and MeterValueSampleInterval is set to 3600 and Power.Active.Import was 300 during last reported Sample, then if Power.Active.Import becomes greater than 500 or less than 100 W, it will automatically send a new Sample.Periodic imediatly and the next Sample.Periodic will be 3600 seconds after this one unless the delta is exceeded again before that or transaction ends.

1.5. UseAuthorizationKeyWithoutDecoding

Required/optional	required
Accessibility	RW
Type	boolean
Description	This configuration key alters the chargers validation and use of AuthorizationKey. When set to false, the charger will attempt to decode the hex-encoded AuthorizationKey before accepting the ChangeConfiguration.req or using it as the password in the http basic auth header. It Will also reject any AuthorizationKey that has an invalid hex-encoding or contains CTL characters or ':' when decoded (see RFC 7617 and appendix of RFC 5234). When set to true, it will use the possibly hex-encoded AuthorizationKey directly as the basic auth header password. It will attempt the same validation as when set to false, but will perform the validation without attempting decoding. Default value is false when AuthorizationKey has not been set prior to the introduction of this configuration key. Otherwise, the value defaults to true.