Device Connector Library External API's

Data Storage

All Fetched credentials will be placed in the Data folder in the program root directory. Amazon Root Ca is by default placed in this folder. The operational certificates will be fetched here. The IoT endpoint will be automatically fetched from this folder for all HTTP and MQTT requests after activation.

Note: It is assumed that all file directories will start at the root program directory.

Data/RootCA.crt should be present before starting program since it is the root ca which is used for establishing the TLS Connection To Activation Portal.

ActivationURI.config should be present which points to the environment of the activation server.

IoT Specific Generated Files

Data/operationalCertificate.pem - The Operational Certificate Fetched From Activation Portal

Data/EndPoint.config - The IoT Endpoint For Communication

Data/CN.config - Client Name Required For MQTT Connection

Csr1.csr - Csr Generated For Connection To Activation Portal

Lora Specific Generated Files

Csr.csr - Csr Generated For Connection To Activation Portal cups.key - Key File generated in device is used as CUPS key

Data/cups.crt - CUPS certificate for communication.

Data/cups.trust - CUPS Trust CA Certificate for communication

Data/URI.config - Endpoint For CUPS

Common API’s

iotsafe_initalize

//Initializes IoT Safe And Connects to activation portal if no operational certificate found,Need to be called once at least once at first.

Usage:

int iotsafe_initialize(int reset ,int mode ,char * EUI);

Example:

iotsafe_initialize(0,0,NULL);

Parameters:

- Reset - 0 for No Reset (Doesn’t Call Activation portal If Certificate Already Present), 1 For Reset (Deletes Existing Credentials and Calls Activation Portal For New Certificates)
- Mode - 0 For Iot, 1 For Lora
- EUI (applicable for Lora Mode ,Left NULL for IOT)- Example: 00:80:00:00:A0:00:8F:65

Returns

- 0 If No Error
- -1 If Error Occured

cleanup_iot

//Clean ups temporary files , should be called only before the program exits.Only after all communication has taken place.

Usage:

cleanup_iot()

Returns

- 0 If No Error

IoT Specific API’s

http_req()

//Sends http request within the parameters specified and returns the response.

Usage:

http_req(char * path,int port, char *message,char * buffer,int buffer_len)

Example :

char buffer[512];

http_req(“temperature/end”,443,”Hello From Http”,buffer,512);

Parameters:

- Path - Suppose endpoint is www.google.com/temperature/end , then path is “temperature/end” , the first / can be ignored.
- Port- Port should be given for endpoint
- Message - Message should be present if Post request, If NULL then Get Request is sent
- Buffer - Buffer to Save response to
- Buffer_Len - Maximum Length Of Buffer Array

Returns

- -2 If Certificate Is Possibly Deactivated.
- -1 If http Connection Setup has Error
- 0 If Connection And Request Success (200 Status returned by server)
- >0 if Connection success but request has error code .Example Returns 403 (When error 403 is returned by server)

iotsafe_getMQTTConnection

//To set up an mqtt connection to the operational endpoint.

Usage:

iotsafe_getMQTTConnection(int port);

Example :

iotsafe_getMQTTConnection(7778);

Parameters:

- Port- Port should be given for endpoint

Returns

- -1 If Http Connection Setup Has Error
- 0 If Connection Is Established

iotsafe_publish

// Used to Publish Message to an mqtt topic .Must be called after getting connection.

Usage:

iotsafe_publish(char * message,char * Topic);

Example :

iotsafe_publish("Hi From Mqtt","Temperature");

Parameters:

- Message- Message To be Sent
- Topic - Topic to sent message to

Returns

- -1 If Publish has Error
- 0 If Published Success

iotsafe_disconnect

// Used to Disconnect from an mqtt connection.

Usage:

iotsafe_disconnect()

Example :

iotsafe_disconnect()

Returns

- -1 If Disconnect Error
- 0 If Disconnect Success

Multitech Specific APi

fetch_eui

// To be run on multitech devices. Fetches the device eui from “/sys/devices/platform/mts-io/lora/eui” and passes to buffer

Usage:r

fetch_eui(char * buffer)

Example :

char eui[50];

fetch_eui(eui);

Returns

- 0 If Fetch Success
- -1 If Fetch Fail

export_lora

// Fetches the credentials received from activation portal and updates multitech device with these credentials.Requires loraconfig.json in root folder,this contains the basic station parameters customizable by the user.

Usage:

export_lora

Example :

export_lora()

Returns

- 0 If Export Sucess
- -1 If Export Fail