TEST for PDF

Appendix C: ENGAGE™ Persistent Connection API

The ENGAGE Gateway implements a general 30 second timeout for all commands below; it is suggested that Alliance Partners use a similar value when implementing a timeout on Gateway responses. In particular, the /edgeDevices/{linkId}/database resource can take many seconds to complete for large databases which must be first transferred and then queued by the Gateway before a response is returned to the initiating client.

Due to the Gateway's webserver implementation, extraneous forward slashes (/) used in endpoint URLs shown below are ignored; while this would be a non-standard use case in a machine-to-machine developed solution, it is noted herein for clarity's sake.

The reader-controller shares many of the same resources as the Gateway. The key difference being that the reader-controller is not linked to any other edge devices. As a result, the reader-controller will utilize the /edgeDevices/{linkId}/ resources, but the linkId will always be dev00000.

Persistent Connection API URIs	Device Applicability
Gateway Configuration & Setup	
GET /gateway/newCredentials	GWE
PUT /gateway/newCredentials	GWE
GET /gateway/time	GWE
PUT /gateway/config	GWE
GET /gateway/deviceInfo	GWE
GET /gateway/scanList	GWE
Linking and Controlling Edge Devices	
POST /edgeDevices	GWE
GET /edgeDevices/linkList	GWE
DELETE /edgeDevices/{linkId}	GWE

Persistent Connection API URIs	Device Applicability
PUT /edgeDevices/lockControl	GWE
PUT /edgeDevices/{linkId}/lockControl	GWE, RC
GET /edgeDevices/lockStatus	GWE
GET /edgeDevices/{linkId}/lockStatus	GWE, RC
PUT /edgeDevices/{linkId}/database	GWE, RC
DELETE /edgeDevices/{linkId}/database	GWE, RC
GET /edgeDevices/{linkId}/dbDownloadStatus	GWE, RC
PUT /edgeDevices/{linkId}/config	GWE, RC
DELETE /edgeDevices/{linkId}/config	GWE, RC
GET /edgeDevices/audits	GWE
GET /edgeDevices/{linkId}/audits	GWE, RC
DELETE /edgeDevices/audits	GWE
DELETE /edgeDevices/{linkId}/audits	GWE, RC
GET /edgeDevices/{linkId}/time	GWE, RC
POST /edgeDevices/{linkId}/lastCredential	GWE, RC
GET /edgeDevices/{linkId}/newCredentials	GWE, RC
PUT /edgeDevices/{linkId}/newCredentials	GWE, RC
Gateway Diagnostics	
GET /gateway/gatewayNetworkStatistics	GWE
GET /gateway/gatewayEventLog	GWE

Gateway Configuration and Setup

GET /gateway/newCredentials

This resource is used to initialize basic authentication between IP host and the ENGAGE Gateway. Every GET request results in a new password being randomly generated, and previous randomly generated passwords are forgotten.

This resource is only available after commissioning and before accepting a username/password pair.

Accepting the username/password pair (using PUT /gateway/newCredentials), causes this resource to disappear from the Gateway API. An error response is given if an attempt to access this API resource is executed after the username/password pair has been accepted.

Default username: EngageGatewayDefaultUser

Default password: EngageGatewayDefaultPassword

Authentication

Basic Authentication: Default username and password.

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- credentials = parent
  - If successful, this parameter, and children are included.
- usr = string
  - Proposed username to be used by client. Format: “gateway_(8-byte serial number)
- pwd = string
  - Proposed password to be used by client. Randomly generated password for every request.

Error Response

HTTP status code 403

Response parameters:

Type: application/json

- statusMessage = “Credentials already established”

NOTE: 403 status is only used if the unique credentials are used. If default username/credentials are used, a 401 response is returned.

Example:

```
{"credentials":{"usr":"gateway_a0b100000000003f","pwd":"eh1b1yeDuWdnHZXnYjLxN4yRdkll158f"}}
```

PUT /gateway/newCredentials

This resource is used to initialize basic authentication between IP host and the ENGAGE Gateway. This should be used sequentially after a GET against the same resource. If multiple GETs are done prior to PUT, the Gateway will use the last password generated.

This resource is only available after commissioning and before accepting a username/password pair.

Accepting the username/password pair, causes this resource to disappear from the Gateway API.

Default username: EngageGatewayDefaultUser

Default password: EngageGatewayDefaultPassword

Authentication

Basic Authentication: Default username and password. Please note this is not the usr:pwd pair that was returned in the body of GET /gateway/newCredentials. All future Gateway operations will use the unique username and password pair which the Gateway generated.

Request

URI Parameters

- None

Success Response

HTTP status code 201

Response parameters:

Type: application/json

- statusMessage = “Gateway username / password changed”

Error Response

HTTP status code 403

Response parameters:

Type: application/json

- statusMessage = “Credentials already established”

NOTE: – 403 status is only used if the unique credentials are used. If default username/credentials are used, a 401 response is returned.

GET /gateway/time

This resource returns the time of the Gateway and network of linked devices.

Authentication

Basic Authentication: unique credentials defined and accepted during setup.

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- gatewayTime = string, parent tag to follow
  - rtcTime = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Configuration”
  - dstEnable = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Configuration”
  - dstStart = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Configuration”
  - dstEnd = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Configuration”

Example:

```
{"gatewayTime":{"rtcTime":"20150720142459","dstEnable":"true","dstStart":"3022","dstEnd":"B012
"}}
```

PUT /gateway/config

This resource writes new configurations to the ENGAGE Gateway. The JSON object *gatewayConfig* is used, as defined in the **ENGAGE - JSON Data Structures** document.

Authentication

Basic – unique credentials defined and accepted during setup.

Request

URI Parameters:

Type: application/json

- gatewayConfig = parent tag + children tags, definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Configuration”

Success Response

HTTP status code 200

Response parameters:

- None

Example:

```
{"gatewayConfig":{"genGatewayConfig":{"deviceName":"new GW
Name","rtcTime":"20150720140000","dstEnable":"true","dstStart":"3022",
"dstEnd":"B012","fwurl":"","fwDwnldTm":"","fwImplTm":""},"gatewayIpModeConfig":{"discoveryMeth
od":"zeroConf","fixedIpAddr":"",
"defGatewayIpAddr":"","netmask":"","ipDnsAddr":"","altDnsAddr":""}}}
```

GET /gateway/deviceInfo

This resource is used to retrieve Gateway configurations and read-only parameters. The JSON object *gatewayDeviceInfo* is returned, as defined in the **ENGAGE - JSON Data Structures** document.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response Parameters:

Type: application/json

- gatewayDeviceInfo = parent tag + children tags, definition in the the **ENGAGE - JSON Data Structures** document, section titled “Device Information Records (Gateway)”

Example


```
{"gatewayDeviceInfo":{"fwVer":{"main":"1.49.10","ble2":"0.0.7","mainBl":"2014.10.0.2","macAdr"
:"00409D63028E",
"ethernetMacAdr":"00409D63028D"},"wifiStngs":{"wifiEnable":"","ssid":"","psswd":"","usrNm":"",
"wifiSec":"","kyIndx":""},
"gatewayParameters":{"hostProtocol":"ipEngage","deviceName":"Gateway IP
Mode","mdl":"GWE","mainSn":"a0b1000000000206",
"mfgDt":"20150515","hwVer":"01","daysInUse":"00031","gwUpTime":"017","rtcTime":"20170109134639
","sshEnable":"false",
"dstEnable":"true","dstStart":"3022","dstEnd":"B012","fwurl":"","fwDwnldTm":"","fwImplTm":""},
"ipv4Interfaces":
{"ethernetProperties":{"assignedIpAddr":"10.44.95.46","defGatewayIpAddr":"10.44.92.1","netmask
":"255.255.252.0"},
"wifiClientProperties":{"assignedIpAddr":"","defGatewayIpAddr":"","netmask":""}},"gatewayRsiMo
deConfig":
{"rsdAddress":"","lowDoor":"","highDoor":""},"gatewayIpModeConfig":{"discoveryMethod":"dhcp","
fixedIpAddr":"",
"defGatewayIpAddr":"","netmask":"","ipDnsAddr":"","altDnsAddr":"","ipServerURL":"","caServerUR
L":"","wsKeepAlive":"0"},
"gatewayFeatureList":["rsiMode","ipServerMode","ipClientMode","lockFwUpgrade"],"edgeDeviceFwUr
ls":
[{"fwurl":"","mdl":"nde","fwDwnldTm":"","fwImplTm":"","fwVer":"","status":{"fwDwnld":"","fwUpd
ate":""},"extendedStatus":
[{"linkId":"dev00000","deviceId":"a100000000002479","deviceName":"testNDE","mdl":"nde","fwVer"
:"02.08.12","fwUpdateStatus":"",
"fwUpdateStartTm":"","fwTransferCmpltPercentage":0,"fwUpdateCmpltTm":"","fwUpdateCancelTm":"",
"retries":0}]}]}}
```


GET /gateway/scanList

Issuing a GET against this resource triggers a BLE scan of nearby advertising ENGAGE devices. Every GET triggers a real time scan which utilizes Gateway BLE radio, so it takes a moment for scanning to complete. Issuing too many scans back to back is not recommended.

A scan is required ***prior*** to posting to the edgeDevices resource as described in the **Linking and Controlling Edge Devices** Section.

In heavy 2.4GHz traffic environments and depending on the advertising rates of ENGAGE edge devices, some devices might not be returned in the scan list, so a second request might be needed.

The response is the *gatewayScanList* JSON object defined in the **ENGAGE - JSON Data Structures** document.

If the Gateway has the maximum number of devices linked, it will be unable to scan for new devices to link to, and an error will be returned.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response Parameters:

Type: application/json

- gatewayScanList = parent tag + children tags, definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Scan List”

Error Response(s)

HTTP status code 409

Response parameters:

Type: application/json

- statusMessage = “Linked devices at maximum”

HTTP status code 500

Response parameters:

Type: application/json

- statusMessage = “Request timed out”

Example:


```
{"gatewayScanList":[{"mainSn":"a100000000001234","deviceName":"Front
Door","signalQuality":"Med","modelType":"nde"},
```

{"mainSn":"a100000000005678","deviceName":"Storage Closet","signalQuality":"Med","modelType":"nde"}]}

Linking and Controlling Edge Devices

POST /edgeDevices

Posting to this resource is a request to create a link between the Gateway and the targeted edge device using BLE.

As a pre-requisite, the scanList resource (see the **GET /gateway/scanList** Section) must be accessed prior to this. Using the scanList resource forces the Gateway to do an active BLE scan. Posting to this resource is dependent on a previous successful scan, and the target device was included in the scan list.

The body of this resource must contain the deviceID JSON tag which specifies the serial number of the intended edge device. This serial number should be the same as was reported from the scanList JSON tag “mainSn”.

NOTE: Linking a device to the Gateway is a lengthy process and a delay must be inserted between the linking operation (after reception of a response from the Gateway to this command) and the sending of another command to the newly linked device. This delay should be at least 10 seconds to account for worst-case conditions and to avoid an error being returned by the Gateway in response to the subsequent command.
NOTE: This resource is not applicable to the reader-controller.

Authentication

Basic – unique credentials defined & accepted during setup

Request

URI Parameters:

- None

Body:

Type: application/json

- deviceId = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”

Example:

{"deviceId":"a000000000000123"}

Success Response

HTTP status code 201

Response Parameters

Type: application/json

- linkInfo = parent tag

  - linkId = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - deviceId = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - deviceName = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - linkCommStatus = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - modelType = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”

Example:

```
{"linkInfo":{"linkId":"dev00001","deviceId":"a000000000000123","deviceName":"Front
Door","linkCommStatus":"Connected",
"modelType":"nde"}}
```

Error Response(s)

HTTP status code 400

Response parameters:

Type: application/json

- statusMessage = multiple detailed status message, value(s) defined below

  - “Invalid device ID in JSON body” indicates, bad device ID in JSON body. This error response can also be given if the deviceId contained in the body is already liked to the respective Gateway.
    HTTP status code 500

    Response parameters:

    Type: application/json

- statusMessage = multiple detailed status message, value(s) defined below

  - “Link failed to connect” indicates, request to link to edge device that was not in the “scanList” response

GET /edgeDevices/linkList

This resource retrieves the list of currently linked edge devices. The response contains the JSON object *linkList* as defined in the **ENGAGE - JSON Data Structures** document.

NOTE: The tag “linkCommStatus” should be used to determine if the edge device is currently connected to the Gateway.
NOTE: This resource is not applicable to the reader-controller.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- linkList = parent tag, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”

Example:


```
{"linkList":[{"linkId":"dev00001","linkCommStatus":"Connected","deviceName":"Front
Door","modelType":"nde",
"deviceId":"a000000000000123","macAddr":"00:01:02:03:04:05","macType":"public","battV":{"main"
:"5.23","li":"0"}}]}
```


PUT /edgeDevices/lockControl

This resource sends a broadcast lock control command to all locks in the linkList that support the lock control command issued. This command puts the edge device into the commanded state until a recalculation of state event occurs (unless the commanded state is a frozen state). This recalculation of state event occurs whenever a special credential (lockdown, toggle, etc.) is presented or a holiday, auto-unlock, or time change is sent to the edge device.

The body of the command is the JSON object *lockControl* as defined in the **ENGAGE - JSON Data Structures** document.
NOTE: When sending lockControl HTTP requests, edge device audits should be consulted to ensure the edge device was able to receive the request and has transitioned into the requisite state.
NOTE: This resource is not applicable to the reader-controller.

Change in functionality for URI:

Gateway FW versions: Prior to 1.24: If the edge devices are configured with their own access control database and they are not wirelessly connected

to the Gateway when this command is issued, the Gateway does not store the lock control command.
Gateway FW versions: 1.24 to latest: Lock state change commands are now cached in the Gateway and delivered upon reconnect to the edge device.
Exception: Momentary unlock commands are not cached in the Gateway. Momentary unlock messages can only be delivered when edge device is connected to the Gateway.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Body

Type: application/json

- lockControl = parent tag + children tags, definition in the **ENGAGE - JSON Data Structures** document, section titled “Device Control” related to the Gateway structures

Example:

```
{"lockControl":{"lockState":{"nextLockState":"frozenSecure"}}}
```

Success Response

HTTP status code 200

Response parameters:

- None

HTTP status code 202

Type: application/json

- statusMessage = “Command cached for, deviceId: [(list)]”, indicates that one or more of the edge devices are currently disconnected from the Gateway. The list is contained in the ordered array.

Error Response(s)

HTTP status code 400

Response parameters:

Type: application/json

- statusMessage = multiple detailed status message, value(s) defined below
  - “Relock delay out of bounds” indicates, relock delay sent in JSON body is out of bounds
  - “Invalid JSON tag” indicates, invalid tag name sent in JSON body
  - “Invalid JSON tag value”, indicates, invalid tag value in JSON body

  HTTP status code 409

  Response parameters:

  Type: application/json

- statusMessage = “Lock(s) offline, momentary unlock command is not executed for, deviceId[(list)]”, indicates momentary unlock command sent to locks currently connected, however momentary unlock messages are not cached.

GET /edgeDevices/lockStatus

This resource is used to retrieve current status of all edge devices.

NOTE: This resource is not applicable to the reader-controller.
The body of the response is the JSON object *lockStatus* as defined in the **ENGAGE - JSON Data Structures** document.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- edgeDevice = parent tag, array containing ordered list of status for each linkId. Each array entry scheme as follows:
  - linkId = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - sensorStatus = parent tag + children; as defined in the **ENGAGE - JSON Data Structures** document, section titled “Lock Status” related to the Gateway structures

- lockStatus = parent tag + children; as defined in the **ENGAGE - JSON Data Structures** document, section titled “Lock Status” related to the Gateway structures

Example:


```
{"edgeDevice":[{"linkId":"dev00001","sensorStatus":{"rexActive":"F","fdrActive":"F","tamperOpe
n":"T","magTamperDetected":"T",
"rtcBatteryStatus":"normal","locked":"T","primaryBatteryStatus":"normal","doorOpen":"F"},"lock
Status":{"currentLockState":"sec"}},
{"linkId":"dev00003","sensorStatus":{"rexActive":"F","fdrActive":"F","tamperOpen":"F","magTamp
erDetected":"F","rtcBatteryStatus":
"normal","locked":"T","primaryBatteryStatus":"normal","doorOpen":"T"},"lockStatus":{"currentLo
ckState":"frznSec"}}]}
```


DELETE /edgeDevices/{linkId}

This resource is used to delete an existing link. The linkId can be retrieved using the /edgeDevices/linkList resource.

Deleting the link at the Gateway does not notify the edge device, other than the BLE connection being disconnected. Without further action by the administrator the edge device behaves as if it has lost a BLE connection with the Gateway.

NOTE: This resource is not applicable to the reader-controller.

Authentication**

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”

Success Response

HTTP status code 200

Response parameters:

- None

PUT /edgeDevices/{linkId}/database

This resource is used to selectively push an access control database change to a specified linked edge device. The payload is the JSON object *db* as defined in the **ENGAGE - JSON Data Structures** document.
Gateway:

The Gateway sends anything that is in the body to the linked edge device. That means the Gateway does not error check the contents of the database. Error checking is done by the edge device, so audit trails should be checked to validate database updates.

A successful response indicates that the edge device has accepted a single test message and the Gateway will begin sending the content to the target edge device. It does not indicate that the message has been completely received by the target edge device.

Overall transfer status can be monitored with the resource /edgeDevice/{{linkId}}/dbDownloadStatus.

Depending on the size of the database, this operation can take considerable time as the maximum transfer rate of the Gateway to a device over BLE is 768 bytes/sec. As such, it is suggested that *db* data be 'minified' by removing all whitespace from the JSON. As an example, a 5000 user database having a size of 984K would have a best case transfer time of ~22 minutes; performing this test in an office environment where there are many 2.4GHz devices and access points competing for bandwidth, the transfer time has been measured to typically be ~50 minutes.
NOTE: An edgeDevice stores the database information into non-volatile memory. It is therefore not advised that database transfers be performed on a highly frequent periodic basis; rather, it is ideal to only send a new database when a change has occurred. Up to 500 "delta" add/mod/delete *db* operations can be sent as well before a complete database should be sent again with all credential entries. This avoids dominating available bandwidth continuously and reduces transfer times.
If the edge device is not connected when this resource is requested the Gateway will not store the message. (See **NOTE:** below)
NOTE: For Gateway firmware version 1.49.24 and later, NDE firmware version 2.8.16 and later, and LE firmware version 2.09.05 and later, this resource supports a database "resume" functionality. If the edge device is disconnected either when this resource is requested, or at any time during the database download, the Gateway attempts to restart the database transfer from the last packet that was successfully sent, once the respective edge device is re-connected. This process is serialized, therefore for a Gateway which has multiple linked edge devices, the device-0 database download will progress until it either completes or disconnects, at which time the device-1 database download will begin and progress until it either completes or disconnects. This round-robin process continues until all the databases have been sent to their respective edge devices.

If a database change must be made immediately to a linked edge device and the Gateway is currently transferring a database to another linked edge device, the resource PUT /edgeDevices/{linkId}/config can be utilized to "jump the line" and send the request immediately to that edge device. Details of how to implement the config endpoint are shown below, with the notable

exception that the JSON body message would contain the “db” JSON tag instead of the “config” JSON tag if used for this purpose.

NOTE: While this is allowed, it is **NOT** a recommended method to be implemented programmatically. In addition, the JSON body size limit for the config endpoint still holds true when sending a database through the config endpoint. For this reason, it is not recommended that the config endpoint be utilized for modifying databases with more than one user addition, modification, or deletion at a time.

Consideration should be given to canceling a database download currently in que, in the event that the config endpoint must be utilized to modify the database door file.

While the “status” tag has been added to the /dbDownloadStatus endpoint to indicate the status of the database resume functionality, edge device audits should still be checked at the completion of the database download to ensure that the database has been properly received and processed by the edge device.

For firmware versions 1.51.22 and later, the Gateway has implemented a 36 hour timeout on database messages that are not able to be transmitted successfully to the edge device. Please check the edge device audits for successful processing of the JSON body.

Reader-Controller:

A successful response indicates that the reader-controller has received the message, but not that the reader-controller has activated the database.

Overall transfer status can be monitored with the resource /edgeDevice/dev00000/dbDownloadStatus.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List” For reader-controllers this is a constant dev00000.

Body:

Type: application/json

- db = parent tag + children, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Lock JSON Messages” in sections defining the related components of the “db” tag.

Example:


```
{"db":{"usrRcrd":{"deleteAll":1,"delete":[],"update":[],"add":[{"usrID":"17856","adaEn":0,"act
DtTm":"20150303000000",
"expDtTm":"21350101000000","fnctn":"norm","crSch":[1],"primeCr":"bc477b6b51700a797fda79646cafb
160","prCrTyp":"card",
"scndCr":null,"scndCrTyp":null}]},"schedules":[{"days":["Su","Mo","Tu","We","Th","Fr","Sa"],"s
trtHr":0,"strtMn":0,"lngth":1439}],
"holidays":[],"autoUnlock":[]},"nxtDbVerTS":"0x8d22651fb4300a2"}
```


Success Response

HTTP status code 202
NOTE: indicates the request was accepted by Gateway or reader-controller and transfer will begin. A 202 does not indicate that it has been delivered to the target edge device.

Error Response(s)

HTTP status code 503

Response parameters:

Type: application/json

- statusMessage = “Currently processing another database”, indicates Gateway, edge device, or reader-controller is processing another database.

DELETE /edgeDevices/{linkId}/database

This resource is used to selectively cancel an access control database change to a specified linked edge device that is currently either processing or in queue to be downloaded. For Gateway firmware version 1.49.24 and later, NDE firmware version 2.8.16 and later, LE firmware version 2.09.05 and later, and all other ENGAGE devices.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”. For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

The database download which was previously in progress has been canceled.

For ENGAGE Gateway Firmware version 1.50.08 only:

The Gateway will report HTTP status code 200 if the database transfer has already been completed and processed by the edge device, or if no database transfer has been initiated yet.

Response parameters:

Type: application/json

- none

Error Response

HTTP status code 403

Database transfer cannot be canceled. Please see the following response options:

Response parameters:

Type: application/json

- “Currently processing previous database”
- “Database is not in progress”

For ENGAGE Gateway Firmware version 1.50.08 only:

HTTP status code 503

The database download cannot be canceled because the edge device has received the database completely and is currently processing the database.

Response parameters:

Type: application/json

- none

GET /edgeDevices/{linkId}/dbDownloadStatus

This resource is used to return the status of a previous edge device database transfer which was requested with the resource /edgeDevices/{linkId}/database.

Supported by Gateway FW versions 1.38 to latest and all other ENGAGE devices

Gateway FW Versions **1.49.24 to latest:** and all other ENGAGE devices: Include an additional JSON tag “status”

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”. For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- downloadStatus = parent JSON object
- “status” = string (GWE v1.49.24 to latest and all other ENGAGE devices)
  - “pending”: Database download to this edge device (linkId) is pending and will be initiated shortly. (Not applicable to reader-controller). This can be caused by either:
    - Database download to another edge device may have been in progress. The Gateway now serializes database transfers to edge devices and transfers databases to one edge device at a time.
    - If this resource is requested immediately after issuing PUT /edgeDevices/{linkId}/database
  - “inProgress”: The Gateway is currently transferring the database to this edge device, or the reader-controller is currently transferring the database.
  - “suspended”: The database download to this device is suspended but will be resumed soon. This can be caused by the edge device disconnecting from the Gateway during a database transfer.
  - “failed”: Something went wrong with the transfer and the IP Host should re-try the database download request.
  - “completed”: The database update to the edge device has completed successfully.
  - “canceled”: The IP host has canceled the most recent database download request.
  - “”No previous status available
- downloadPercentComplete = integer
- Range 0 to 100, indicates current transfer percentage of database from Gateway to the edge device, or from the host to the reader-controller. The value is reset after a new transfer request has been made and accepted by Gateway or reader-controller.

- 0 indicates transfer has not started.
- Value of 100 indicates transfer is complete, but not yet processed by edge device; or that the transfer and process is complete for reader-controllers.
- Value of -1 indicates there has been an interruption in transfer to edge device. Host should retry database update request using /edgeDevices/{linkId}/database
- processPercentComplete = integer (not applicable to reader-controllers)
- Range 0 to 100, indicates processing status of edge device after transfer of database from Gateway to edge device. The value is reset after a new transfer request has been made and accepted by the Gateway.
- 0 indicates the device has not yet begun to process the database update, or the transfer is not yet complete.
- 100 indicates the processing is complete, and the host should request updated audits from edge device to confirm database processed correctly.

Example:

1.38 to 1.48:

```
{"downloadStatus":{"downloadPercentComplete":58,"processPercentComplete":0}}
```

1.49.24 to latest:

```
{"downloadStatus":{"status":"inProgress","downloadPercentComplete":58,"processPercentComplete"
:0}}
```

PUT /edgeDevices/{linkId}/config

This resource is used to selectively push configuration changes to a specified linked edge device. The payload is the JSON object *config* as defined in the **ENGAGE - JSON Data Structures** document.
Gateway:

The Gateway sends anything that is in the body to the linked edge device. This means that the Gateway does not error check the contents of this change. Error checking is done by the edge device, so the audit trails should be checked to validate configuration updates.

For firmware versions 1.50.08 and later: if the edge device is not connected when the command is sent, the Gateway caches the message and tries to send it once the device reconnects. Firmware versions prior to this do not store the message if the device is disconnected when the HTTP request is received.

This resource has a limit on the JSON body message size of 2kb. Minified JSON corresponding to the “config” tag as defined in the **ENGAGE - JSON Data Structures** document should fall well within this limit.

For firmware versions 1.51.22 and later, the Gateway has implemented a 36 hour timeout on config messages that are not able to be transmitted successfully to the edge device. Check the edge device audits for successful processing of the JSON body.

Reader-Controller:
This resource does not have a limit on the JSON body message size, but would accept JSON body messages in 2kb blocks, if desired.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled *Gateway Edge Device Link List.* For reader-controllers this is a constant dev00000.

Body:

Type: application/json

- config = parent tag + children as defined in the **ENGAGE - JSON Data Structures** document, section titled *Device Parameters*, sub-section *Modifiable Parameters*.

Example:


```
{"config":{"model":"","relock":3,"doorProp":20,"ada":30,"firstManIn":0,"dstEnable":1,"dstStart
":"3022","dstEnd":"B012",
"battFail":"sec","bprEn":"T","rtcTime":null,"proxConfHID":"T","proxConfGE4001":"T","proxConfGE
4002":"F","proxConfAWID":"T",
"proxConfGECASI":"T","uid14443":"F","mi14443":"T","noc14443":"T","uid15693":"T","iClsUID40b":"
T"}}
```


Success Response

HTTP status code 200

The JSON body of the response was successfully sent to the requested edge device. Check the edge device audits for successful processing of the JSON body.

HTTP status code 202

The edge device was not connected to the Gateway at the time that the resource was requested. The Gateway will continue to attempt to send the

message once the device re-connects. Check the edge device audits for successful processing of the JSON body. Not applicable to reader-controllers.

Error Response(s)

HTTP status code 500

The Gateway was unable to process the request.

HTTP status code 503

The Gateway is already sending another config or a database to the selected device. Consider canceling the other operation.

Response parameters:

- None

DELETE /edgeDevices/{linkId}/config

This resource is used to selectively cancel an edge device configuration change to a specified linked edge device that is currently either processing or in queue, to be downloaded. For Gateway firmware version 1.51.22 and later and all other ENGAGE devices.

NOTE: Transfers from the Gateway to the edge device from PUT /edgeDevices/{linkId}/config are sent to the edge device in 1024-byte segments. Because of the JSON body size limitation of 2k for PUT /edgeDevices/{linkId}/config if the Gateway has already started the JSON transfer to the edge device when this resource is requested, it is not probable that the message transfer will be able to be interrupted. This resource will be best utilized when the Gateway has queued the JSON transfer to the respective edge device. (i.e., if the Gateway is currently transferring a config to linkId 0004 and this resource is requested for linkId other than 0004, this resource will be most effective)

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the ***ENGAGE - JSON Data Structures** document, section titled *Gateway Edge Device Link List.* For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

The config download which was previously in progress has been canceled by the Gateway or the config download for the reader-controller has been canceled.

Response parameters:

Type: application/json

- none

Error Response

HTTP status code 403

Config transfer cannot be canceled. Please see the following response options:

Response parameters:

Type: application/json

- “Currently processing previous config”
- “Config is not in progress”

GET /edgeDevices/{linkId}/params

This resource is used to selectively request all of the parameters from a single edge device. The response is defined by the JSON object *edgeDevice* in the **ENGAGE - JSON Data Structures** document.
Gateway:

The returned result is a cached version from when the last time the edge device parameters were retrieved by the Gateway. Any change in configuration results in this information being refreshed.

Returned results are defined by the edge device and are unique to the model of edge device.

Reader-Controller:

The results returned by the reader-controller are the current state of each of the parameters.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in **ENGAGE - JSON Data Structures** document, section titled *Gateway Edge Device Link List.* For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- edgeDevice = parent tag and children, as defined in the **ENGAGE - JSON Data Structures** document, subsection titled *Linked Edge Device Data*.

Example:


```
{"edgeDevice":{"linkId":"dev00001","config":{"battV":{"main":"5.23","li":"0"},"fwVer":{"lock":
"02.04.17","main":"02.04.17",
"mainBl":"01.04.00","credRdr":"02.00.29","credRdrBl":"01.02.09","ble":"01.03.108","wifi":"4.62
.r1606","macAdr":
"00:01:02:03:04:05"},"lockPrmtrs":{"name":"Front
Door","mdl":"unknown","type":"strm","lckMode":"410ip","lckSn":
"a000000000000123","mainSn":"a000000000000123","hwVer":"00","daysInUse":242,"mfgDt":"ffffffff"
,"relock":7,"doorProp":20,
"ada":30,"firstManIn":0,"dstEnable":255,"dstStart":"ffff","dstEnd":"ffff","battFail":"sec","rt
cTime":"20150814162605",
"dbDwnldTm":"200000000000","fwUrl":"","fwDwnLdTm":"200000000000","fwImpTm":"200000000000"},"rd
rPrmtrs":{"bprEn":"T","sn":
"1112131415161718","hwVer":"00","mfgDt":"191a1b1c","configCrdPrsntd":"F","proxConfHID":"T","pr
oxConfGECASI":"T","proxConfGE4001":
"T","proxConfGE4002":"F","proxConfAWID":"T","iClsUID40b":"T","uid14443":"F","mi14443":"T","noc
14443":"T","uid15693":"T",
"iCls15693":"F","iClsFrmt":"disbld","pivCnfg":"75bPIV"}}}}
```


GET /edgeDevices/{linkId}/audits

This resource is used to retrieve all the audits received by the linked edge device. Audit structure is defined in the **ENGAGE - JSON Data Structures** document and the meaning of audits are defined in the **ENGAGE - Audits** document.
Gateway:

The audits are pushed in real-time from the edge device to the Gateway. Once transferred to the edge device it is expected that the audits are removed from the edge device and reside only inside the Gateway memory. It is possible however that an audit may be reported to the Gateway twice if the edge device was disconnected from the Gateway before the communication of audits was completed. Alliance partners may wish to filter these duplicate messages if the display of duplicate audits is undesirable.

NOTE: It is important to note that the Gateway does not validate information sent through it, intending to operate as a store and forward bridge as much as possible, and thus the audits generated in response to

operations by the edge device itself should be consulted to validate proper processing occurred.

In addition, it is recommended that our Alliance Partners store at least the last 30 days of all generated audits from ENGAGE devices in order to provide verification of events.

Reader-Controller:
The reader-controller posts audits to the host, when connected, in real-time. This resource can be used to retrieve audits that were not sent when the device was disconnected from the host.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”. For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- edgeDevice = parent tag, object containing ordered list of audits as follows:
  - linkId = definition in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”
  - audits = parent tag array, as defined in the **ENGAGE - JSON Data Structures** document
  - overwriteIndex = zero based index for the location of the “oldest” audit. This tag only appears after the Gateway has received more than 200 audits from the edge device without the audits being requested from the IP Host. Not applicable to reader-controllers.

Example:


```
{"edgeDevice":{"linkId":"dev00001","audits":[{"event":"05020000","time":"20150814163754"},{"ev
ent":"000045c0","time":
"20150814163754"},{"event":"0f060000","time":"20150814163800"}]}}
```


DELETE /edgeDevices/{linkId}/audits

Gateway:

Using this resource clears the Gateway's memory for all audits associated with the specified link Id. The Gateway uses an individual circular buffer of audits for each edge device, overwriting oldest entries should the queue of a particular device not be serviced in a timely fashion. The size of this buffer is 2000 audits, the same as a lock.

Change in functionality for URI:

Gateway FW versions: Prior to 1.38: If audits are added after a GET, and a DELETE is issued unread audits may be deleted.
Gateway FW versions: 1.38 to latest: If the audit trail for a given linkId has not been previously requested (GET /edgeDevices/{linkId}) the audit trail will not be deleted.

Reader-Controller:

This resource clears the reader-controller's memory buffer for all audits that it has stored. The reader-controller uses an individual circular buffer of audits, overwriting oldest entries when it reaches 5000 audits.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled "Gateway Edge Device Link List". For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

GET /edgeDevices/audits

This resource behaves as GET /edgeDevices/{linkId}/audits with the notable exception that it retrieves the audits from all of the edge devices linked to the respective Gateway.

NOTE: This resource is only applicable to Gateway firmware versions 1.50.06 and later.
NOTE: This resource is not applicable to the reader-controller.

Authentication

Basic – unique credentials defined & accepted during setup

Request

URI parameters: None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- edgeDevices = parent tag, array containing ordered list of audits per edge device as follows:

  - linkId = definition in the **ENGAGE - JSON Data Structures** document, section title “Gateway Edge Device Link List”
  - audits = parent tag array, as defined in the **ENGAGE - JSON Data Structures** document
  - overwriteIndex = zero based index for the location of the “oldest” audit. This tag only appears after the Gateway has received more than 2000 audits from the edge device without the audits being requested from the IP Host.

Example:


```
{"edgeDevices":[{"linkId":"dev00001","audits":[{"event":"0f040000","time":"20170315124158"}]},
{"linkId":"dev00000",
"audits":[{"event":"0f000000","time":"20170315124113"},{"event":"0f040000","time":"20170315124
205"}]}]}
```


DELETE /edgeDevices/audits

This resource behaves as DELETE /edgeDevices/{linkId}/audits with the notable exception that it deletes the Gateway’s memory of audits for all of the edge devices linked to the respective Gateway.

NOTE: This resource is only applicable to Gateway firmware versions 1.50.06 and later.
NOTE: This resource is not applicable to the reader-controller.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

- None

Success Response

HTTP status code 200

Error Response(s)

HTTP status code 409

Response parameters:

Type: application/json

- statusMessage **=** **“**audits are not deleted for, linkId:[(list)]”, indicates that audits were not deleted for one or more edge devices.

PUT /edgeDevices/{linkId}/lockControl

This resource is a lock control command to a specific lock in the edgeDevice list. This command puts the edge device into the commanded state until a recalculation of state events occurs (unless the commanded state is a frozen state). This recalculation of state events occurs whenever a special credential (lockdown, toggle, etc.) is presented or a holiday, auto-unlock, or time change is sent to the edge device.

The body of the command is the JSON object *lockControl* as defined in the **ENGAGE - JSON Data Structures** document.
NOTE: When sending lockControl HTTP requests, edge device audits should be consulted to ensure the edge device was able to receive the request and has transitioned into the requisite state.

Change in functionality for URI:

Gateway FW versions: Prior to 1.24: If the edge devices are configured with their own access control database and they are not wirelessly connected to the Gateway when this command is issued, the Gateway will not store the lock control command.
Gateway FW versions: 1.24 to latest: lock state change commands are now cached in the Gateway and delivered upon reconnect to the edge device. Exception: momentary unlock commands are not cached in the Gateway. Momentary unlock messages can only be delivered when the edge device is connected to the Gateway

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters:

- None

Body:

Type: application/json

- lockControl = parent tag + children tags, definition in **ENGAGE - JSON Data Structures** document, section titled “Device Control” related to the Gateway structures

Example:

```
{"lockControl":{"lockState":{"nextLockState":"frozenSecure"}}}
```

Success Response

HTTP status code 200

Response parameters:

- None

HTTP status code 202

Type: application/json

- statusMessage = “Command cached”, indicates the target edge device is not currently connected to Gateway, and the message will be delivered upon reconnection. Not applicable to reader-controllers.

Error Response(s)

HTTP status code 400

Response parameters:

Type: application/json

- statusMessage = multiple detailed status message, value(s) defined below
  - “Relock delay out of bounds” indicates, relock delay sent in JSON body is out of bounds
  - “Invalid JSON tag” indicates, invalid tag name sent in JSON body
  - “Invalid JSON tag value”, indicates, invalid tag value in JSON body

HTTP status code 409

Response parameters:

Type: application/json

- statusMessage = “Lock offline, momentary unlock command is not executed for, deviceId[(list)]”, indicates momentary unlock command was not delivered due to the edge device not being currently connected to Gateway. Not applicable to reader-controllers.

GET /edgeDevices/{linkId}/lockStatus

This resource is used to retrieve current status of the specified edge device.

The body of the response is the JSON object *lockStatus* as defined in the **ENGAGE - JSON Data Structures** document.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI parameters:

Type: path parameter

- linkId = linked edge device identifier, as defined in the **ENGAGE - JSON Data Structures** document, section titled “Gateway Edge Device Link List”. For reader-controllers this is a constant dev00000.

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- edgeDevice = parent tag, children tag as follows:
  - linkId = definition in **Schlage ENGAGE - JSON Data Structures** document. For reader-controllers this is a constant dev00000.
  - sensorStatus = parent tag + children; as defined in the **ENGAGE - JSON Data Structures** document, section titled “Lock Control” related to the Persistent Connection structures.
  - lockStatus = parent tag + children; as defined in **ENGAGE - JSON Data Structures** document, section titled “Lock Status” related to the Persistent Connection structures.

Example:


```
{"edgeDevice":{"linkId":"dev00001","sensorStatus":{"rexActive":"F","fdrActive":"F","tamperOpen
":"T","magTamperDetected":"T",
"rtcBatteryStatus":"normal","locked":"T","primaryBatteryStatus":"normal","doorOpen":"F"},"lock
Status":{"currentLockState":"sec"}}}
```


GET /edgeDevices/{linkId}/time

Gateway Firmware Versions 1.52.XX and later.
This resource is used to retrieve the current real-time clock (**rtc**) time from the specified edge device. When this resource is requested the Gateway requests the **rtc** time from the lock, sends this information to the host (in the response to this request), and caches this time for the next 8 hours. If any further requests of this endpoint are issued within 8 hours, the Gateway responds with

the previously cached **rtc** time for that edge device. This 8 hour window prevents consecutive requests from waking the lock to send updated **rtc** time. For further requests after 8 hours the Gateway requests the **rtc** time from the lock, sends this information to the host (in the response to this request), and caches this time for the next 8 hours.
The time resolution reported is in minutes, therefore the seconds field always returns 00. If the edge device is disconnected from the Gateway when this resource is requested the Gateway returns the cached **rtc** value. Therefore it is recommended that the connection status of the edge device be consulted via the linkList resource if the **rtc** value in the response is the same as was sent previously and 8 hours have elapsed since the last request.

Authentication

Basic – unique credentials defined and accepted during setup.

Request

URI parameters:

- None

Success Response

HTTP status code 200

Response parameters:

Type: application/json

- rtcTime = definition in **ENGAGE – JSON Data Structures** document.
  - If the request is received while the edge device is disconnected and the Gateway does not have a cached time value, the returned value will be “”.

Example:

```
{"rtcTime”:”20170405051500}
```

Gateway Diagnostics

The Gateway diagnostics were intended to provide our Software Alliance Members, Integrators, and Installers with additional information about message flow through the ENGAGE Gateway as well as time based network statistics to allow for trouble shooting purposes in sites which are already running.

While these Gateway resources are being published herein, these **SHOULD NOT** be implemented programmatically. Their final design will most certainly change. While one of the core tenants of the ENGAGE platform of interconnected devices, is to maintain backwards compatibility, the design provided in this section **WILL NOT** maintain backwards compatibility until the design has been finalized. These are being published only to provide a

"sneak peek" at what the intention may be for future functionality. This paragraph of text shall be removed from this document once the design has been finalized and is available for programmatic implementation.

Available in Gateway Firmware Versions 1.50.08 and later.

GET /gateway/gatewayNetworkStatistics

****Refer to the above Gateway Diagnostics introduction paragraph regarding programmatic implementation and backwards compatibility.**

This resource is used to retrieve a time based log of network statistics organized by linkId. Results are categorized by the following time frames:

- Quarter Hourly for the past Hour
- Hourly for the past Week
- Daily for the past Month

NOTE:

For each respective edge device, the quarter hourly information (x4) must be fully populated before the first hourly information is populated. In addition, the hourly information (x168) must be fully populated before the first daily information is populated. For example, after a lock has been linked to a Gateway for 3 days, all the quarter hourly information and 72 of the hourly entries will be filled, however none of the daily entries will be filled.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response Parameters:

Type: application/json

- "tx" = The number of transmissions sent to the device
- "rx" = The number of transmissions received from the device
- "rssi" = The mathematical mean of the RSSI values
- "display_time" = the time at which the snapshot was taken

- “disc” = The number of times this device disconnected from the Gateway
- “rec” = The number of times this device re-connected to the Gateway
- “ret” = The number of messages which did not pass the CRC check and had to be retried
- “mac” = The MAC address of the device
- “name” = The friendly name of the device

Example

The JSON body of the return from this request is the equivalent of 29355 lines of JSON (if EOL characters were not removed). Only a small snippet of this JSON is shown below for conciseness.


```
{"json_version":"WNL_V00.01","Wireless Network Log":{"quarter-
hourly":[[{"tx":0,"rx":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":0,"rx
":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":0,"rx
":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":0,"rx
":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":0,"rx
":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":0,"rx
":0,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":0,"mac":"XX:XX:XX:XX:XX:XX","name":"NULL"},{"tx":16,"r
x":80,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":16,"mac":"00:0B:57:1D:58:47","name":"LE-
01"},{"tx":16,"rx":80,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":16,"mac":"00:0B:57:0B:DA:B3","name":"LE-
02"},{"tx":16,"rx":80,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":16,"mac":"00:0B:57:1C:FE:54","name":"LE-
03"},{"tx":16,"rx":80,"rssi":0,"display_time":
"20170314095626","disc":0,"rec":0,"ret":16,"mac":"00:0B:57:1D:51:8D","name":"LE-04"}]],

[…],

"hourly":[[{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"m
ac":"00:07:80:13:E4:22","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"mac"
:"00:07:80:78:2F:80","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"mac"
:"00:07:80:13:93:45","name":
```

```
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"mac"
:"00:07:80:13:9C:E4","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"mac"
:"00:07:80:1F:15:CF","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170314084033","disc":0,"rec":0,"ret":0,"mac"
:"00:07:80:1F:5B:00","name":
"NULL"},{"tx":189,"rx":945,"rssi":-
68,"display_time":"20170314084033","disc":0,"rec":0,"ret":189,"mac":"00:0B:57:1D:58:47",
"name":"LE-01"},{"tx":189,"rx":947,"rssi":-
67,"display_time":"20170314084033","disc":0,"rec":0,"ret":190,"mac":
"00:0B:57:0B:DA:B3","name":"LE-02"},{"tx":188,"rx":943,"rssi":-
73,"display_time":"20170314084033","disc":0,"rec":0,"ret":188,
"mac":"00:0B:57:1C:FE:54","name":"LE-03"},{"tx":188,"rx":943,"rssi":-
75,"display_time":"20170314084033","disc":0,"rec":0,
"ret":188,"mac":"00:0B:57:1D:51:8D","name":"LE-04"}]],

[…],

"daily":[[{"tx":0,"rx":0,"rssi":0,"display_time":"20170307123205","disc":0,"rec":0,"ret":0,"ma
c":"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170307123207","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170307132734","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170307123219","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170307132731","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"20170307132732","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"19700101000000","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"19700101000000","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"19700101000000","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"},{"tx":0,"rx":0,"rssi":0,"display_time":"19700101000000","disc":0,"rec":0,"ret":0,"mac"
:"XX:XX:XX:XX:XX:XX","name":
"NULL"}]]}}
```


GET /gateway/gatewayEventLog

****Refer to the above Gateway Diagnostics introduction paragraph regarding programmatic implementation and backwards compatibility.**

This resource is used to view the message flow into and out of the ENGAGE Gateway.

Authentication

Basic – unique credentials defined and accepted during setup

Request

URI Parameters

- None

Success Response

HTTP status code 200

Response Parameters:

Type: application/json

Example:

```
[05:05:00:01:06:482][dev00000][Dev Connection][Connected]

[05:05:00:01:11:165][dev00000][Dev Info][OK]{"battV":{"main":" 5.81","li":

[05:05:00:01:16:336][dev00000][Data Tx Status][FW Xfer Resumed]
```

Generic Error Responses

Generic error message responses used in the Gateway API:

Error Response

Type: application/json

Response parameters:

- statusMessage = detailed status message, value(s) defined below

HTTP status code 400

statusMessage = “Invalid URI”

HTTP status code 401

statusMessage = “Authorized users only”

Generic error message where the URI includes the linkId URI parameter:

Error Response

Type: application/json

Response parameters:

- statusMessage = detailed status message, value(s) defined below

  HTTP status code 403

  statusMessage = “Device ID does not exist”