dsadas

View | Edit | Delete | Revisions

dsad

Streetlights MQTT API 1.0.0

- [Apache 2.0](#)
- [application/json](#)

The Smartylighting Streetlights API allows you to remotely manage the city lights.

Check out its awesome features:

- Turn a specific streetlight on/off 🌆
- Dim a specific streetlight 😎
- Receive real-time information about environmental lighting conditions 📈

Servers

- mqtt://test.mosquitto.org:{port}/mqttproduction

Test broker

URL Variables › | Expand all

object

port

required

string

Secure connection (TLS) is available through port 8883.

Default value:"1883"

Allowed values:
- "1883"
- "8883"

Security:

- API key
  - In: user

  Provide your API key as the user and leave the password empty.

- OAuth2
  - Required scopes: streetlights:on, streetlights:off, streetlights:dim

  Flows to support OAuth 2.0

  - Flow:Implicit
  Auth URL:https://authserver.example/auth
  Scopes:
    - streetlights:on
    - streetlights:off
    - streetlights:dim
  Flow:Authorization Code
  Auth URL:https://authserver.example/auth
  Token URL:https://authserver.example/token
  Refresh URL:https://authserver.example/refresh
  Scopes:
    - streetlights:on
    - streetlights:off
    - streetlights:dim
  Flow:Client credentials
  Token URL:https://authserver.example/token
  Scopes:
    - streetlights:on
    - streetlights:off
    - streetlights:dim
  Flow:Password
  Token URL:https://authserver.example/token
  Scopes:
    - streetlights:on
    - streetlights:off
    - streetlights:dim
  - Open ID

Operations

- RECEIVE smartylighting/streetlights/1/0/event/{streetlightId}/lighting/measured

  The topic on which measured values may be produced and consumed.

  Inform about environmental lighting conditions of a particular streetlight.

  Operation IDreceiveLightMeasurement

  Available only on servers:

    ○ [production](#)

Parameters ›	Expand all

  object

  streetlightId

  required

  string

  The ID of the streetlight.

Operation specific informationmqtt ›	Expand all

  object

  qos

  1

  Accepts the following message:

  Light measuredlightMeasured

  Inform about environmental lighting conditions of a particular streetlight.

    ○ [application/json](#)

  Message IDlightMeasured

Payload ›	Expand all

  object

  lumens

  integer

  >= 0

  Light intensity measured in lumens.

  sentAt

  string

  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

Headers ›	Expand all

  object

  my-app-header

  integer

  [0 .. 100]

  Additional properties are allowed.

  Examples

Payload ›

  ```
  {
    "lumens": 0,
    "sentAt": "2019-08-24T14:15:22Z"
  }
  ```

  This example has been generated automatically.

Headers ›

  ```
  {
    "my-app-header": 100
  }
  ```
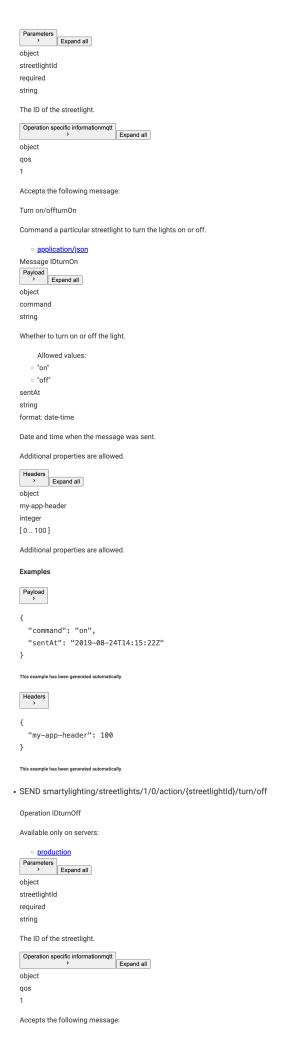
  This example has been generated automatically.

- SEND smartylighting/streetlights/1/0/action/{streetlightId}/turn/on

  Operation IDturnOn

  Available only on servers:

    ○ [production](#)

Parameters > | Expand all

object

streetlightId

required

string

The ID of the streetlight.

Operation specific informationmqtt > | Expand all

object

qos

1

Accepts the following message:

Turn on/offturnOn

Command a particular streetlight to turn the lights on or off.

- application/json

Message IDturnOn

Payload > | Expand all

object

command

string

Whether to turn on or off the light.

  Allowed values:
  - "on"
  - "off"

sentAt

string

format: date-time

Date and time when the message was sent.

Additional properties are allowed.

Headers > | Expand all

object

my-app-header

integer

[0 .. 100]

Additional properties are allowed.

Examples

Payload >

```
{
  "command": "on",
  "sentAt": "2019-08-24T14:15:22Z"
}
```

This example has been generated automatically.

Headers >

```
{
  "my-app-header": 100
}
```

This example has been generated automatically.

- SEND smartylighting/streetlights/1/0/action/{streetlightId}/turn/off

  Operation IDturnOff

  Available only on servers:

  - production

  Parameters > | Expand all

  object

  streetlightId

  required

  string

  The ID of the streetlight.

  Operation specific informationmqtt > | Expand all

  object

  qos

  1

  Accepts the following message:

Turn on/offturnOn

Command a particular streetlight to turn the lights on or off.

- application/json

Message IDturnOn

Payload > | Expand all

object

command

string

Whether to turn on or off the light.

Allowed values:
- "on"
- "off"

sentAt

string

format: date-time

Date and time when the message was sent.

Additional properties are allowed.

Headers > | Expand all

object

my-app-header

integer

[0 .. 100]

Additional properties are allowed.

Examples

Payload >

```
{
  "command": "on",
  "sentAt": "2019-08-24T14:15:22Z"
}
```

This example has been generated automatically.

Headers >

```
{
  "my-app-header": 100
}
```

This example has been generated automatically.

- SEND smartylighting/streetlights/1/0/action/{streetlightId}/dim

Operation IDdimLight

Available only on servers:

- production

Parameters > | Expand all

object

streetlightId

required

string

The ID of the streetlight.

Operation specific informationmqtt > | Expand all

object

qos

1

Accepts the following message:

Dim lightdimLight

Command a particular streetlight to dim the lights.

- application/json

Message IDdimLight
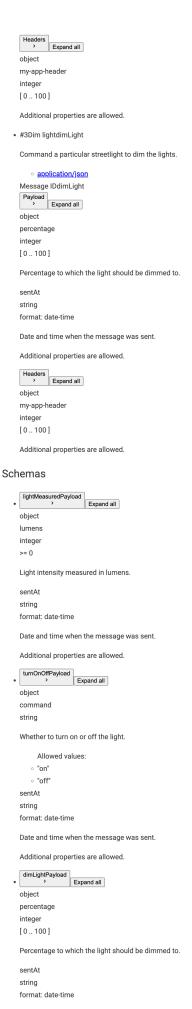
Payload > | Expand all

object

percentage

integer

[0 .. 100]

Percentage to which the light should be dimmed to.

sentAt

string
format: date-time

Date and time when the message was sent.

Additional properties are allowed.

[Headers ›] [Expand all]

object

my-app-header

integer

[0 .. 100]

Additional properties are allowed.

Examples

[Payload ›]

```
{
  "percentage": 100,
  "sentAt": "2019-08-24T14:15:22Z"
}
```

This example has been generated automatically.

[Headers ›]

```
{
  "my-app-header": 100
}
```

This example has been generated automatically.

Messages

- #1Light measuredlightMeasured

  Inform about environmental lighting conditions of a particular streetlight.

  - application/json

  Message IDlightMeasured

  [Payload ›] [Expand all]

  object

  lumens

  integer

  >= 0

  Light intensity measured in lumens.

  sentAt

  string

  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

  [Headers ›] [Expand all]

  object

  my-app-header

  integer

  [0 .. 100]

  Additional properties are allowed.

- #2Turn on/offturnOnOff

  Command a particular streetlight to turn the lights on or off.

  - application/json

  Message IDturnOnOff

  [Payload ›] [Expand all]

  object

  command

  string

  Whether to turn on or off the light.

    Allowed values:
  - "on"
  - "off"

  sentAt

  string

  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

Headers
› Expand all

object

my-app-header
integer
[0 .. 100]

Additional properties are allowed.

- #3Dim lightdimLight

  Command a particular streetlight to dim the lights.

    ○ application/json

  Message IDdimLight

  Payload
  › Expand all

  object

  percentage
  integer
  [0 .. 100]

  Percentage to which the light should be dimmed to.

  sentAt
  string
  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

  Headers
  › Expand all

  object

  my-app-header
  integer
  [0 .. 100]

  Additional properties are allowed.

Schemas

- lightMeasuredPayload
  › Expand all

  object

  lumens
  integer
  >= 0

  Light intensity measured in lumens.

  sentAt
  string
  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

- turnOnOffPayload
  › Expand all

  object

  command
  string

  Whether to turn on or off the light.

      Allowed values:
    ○ "on"
    ○ "off"
  sentAt
  string
  format: date-time

  Date and time when the message was sent.

  Additional properties are allowed.

- dimLightPayload
  › Expand all

  object

  percentage
  integer
  [0 .. 100]

  Percentage to which the light should be dimmed to.

  sentAt
  string
  format: date-time

  Date and time when the message was sent.

Additional properties are allowed.

- sentAt
  string
  format: date-time

  Date and time when the message was sent.