ezlo innovation

mios   vera   fortrezz   centralite

Core events

The Core module notifies a script subscribed by core.subscribe("script_name") about state of the Core module. Structure of an event is table.

Type of events:

---

device_added

sent when device got registered for some gateway

```
{
    event = "device_added"
    _id = "588b7eb528b12d03be86f36f"
    parentDeviceId = "588t7eb528b12d03be86f36f"
    deviceTypeId = "16_4_1_351_8706_256"
    gatewayId = "588b76a44e8c6e50a2826d9f"
    category = "dimmable_light"
    subcategory = "dimmable_bulb"
    name = "_DEVICE_NAME_"
    deviceType = "switch"
    batteryPowered = false,
    reachable = true
    armed = false
    roomId = "_ROOM_ID_"
    security = "low"
    ready = true
    status = "idle"
}
```

Field	Type	Required	Description
event	string	+	type of an event
_id	string	+	an id of the device

parentDeviceId	string	+	an id of the parent device. Empty in case of main device
category	string	+	a device category
subcategory	string	+	a device subcategory
deviceTypeId	string	+	a device type id, generated from manufacturer info
gatewayId	string	+	an id of a gateway this device belongs to
name	string	+	a device name
deviceType	string	+	a device type
batteryPowered	bool	+	is device battery powered
reachable	bool	+	is device reachable
armed	bool	+	is device armed by house mode
roomId	string	+	an id of a room this device is assigned to
persistent	bool	-	is device persistent. Persistent device can't be removed by force removing. False by default.

info	object	-	some additional information for this device
security	string	+	Security level how the device is connected. Possible options: `no`, `low`, `middle`, `high`
ready	bool	+	Ready status of device. `true` value means device is ready to any changes. `false` value means device is busy.
status	string	+	Possible options: `idle`, `broken`. `idle` - device is in normal mode, `broken` - device has invalid data.

device_updated

sent when some changes happened to a device (contains only changes).

```
{
    event = "device_updated"
    _id = "588b7eb528b12d03be86f36f"
    deviceArmed = "588t7eb528b12d03be86f36f"
    name = "switch"
}
```

Field	Type	Required	Description
event	string	+	type of an event
_id	string	+	An id of the device
deviceArmed	bool	+	See device.armed

| __DEVICE_CHANGEABLE_FIELD | any | + |

Device fields which may be updated (__DEVICE_CHANGEABLE_FIELD):

Field	Type	Description
name	string	a device name
reachable	bool	whether device is reachable
roomId	string	an id of a room this device is assigned to
ready	bool	Ready status of device. `true` value means device is ready to any changes. `false` value means device is busy.

---

device_removed

sent when a device got unregistered from some gateway.

```
{
    event = "device_removed"
    _id = "588b7eb528b12d03be86f36f"
    deviceArmed = "588t7eb528b12d03be86f36f"
    name = "switch"
}
```

Field	Type	Required	Description

event	string	+	type of an event
_id	string	+	an id of the device

item_added

sent when an item got registered for some device (can be sent only as a part of a device.added sequence).

```
{
    event = "item_added"
    _id = "588b7eb528b12d03be86f36f"
    deviceId = "588t7eb528b12d03be86f36f"
    enum = {}
    deviceName = "Switcher"
    deviceArmed = false
    hasGetter = true
    hasSetter = false
    name = "alarm_water"
    show = true
    scale = ""
    valueType = int
    valueFormatted = ""
    value = 0
    valueMin = 0
    valueMax = 0
    elementsMaxNumber = 2
    userCodeRestriction = "\d{4}"
}
```

Field	Type	Required	Description
event	string	+	type of an event

_id	string	+	an id of the item
deviceId	string	+	an id of a device this item belongs to
enum	array	-	finite array of possible token values
deviceName	string	-	see device.name !!!MUST BE DROPPED!!! temporarry add for house mode events
deviceArmed	bool	-	see device.armed !!!MUST BE DROPPED!!! temporarry add for house mode events
hasGetter	bool	+	whether the item provides an ability to get a value
hasSetter	bool	+	whether the item provides an ability to set a value
name	string	+	a name(type) of the item
show	bool	+	whether to show the item (on the UI) or not

scale	string	-	a name of measurement units
valueType	string	+	a type of an item's value
valueFormatted	string	+	an item formatted value
value	object	+	an item value
valueMin	object	-	lower limit of item's value field
valueMax	object	-	upper limit of item's value field
elementsMaxNumber	int	-	max allowed elements of a dictionary value
userCodeRestriction	string	-	restriction for a userCode code field http://www.lua.org/manual/5.3/manual.html#6.4.1

item_updated

sent when some changes happened to an item (contains only changes).

```
{
    event = "item_updated"
    _id = "588b7eb528b12d03be86f36f"
```

```
    deviceId = "588t7eb528b12d03be86f36f"
    deviceName = "Switcher"
    deviceArmed = false
    deviceCategory = "switch"
    deviceSubcategory = "in_wall"
    roomName = ""
    notifications = { "12314324", "978343" }
    name = "alarm_water"
    elementsMaxNumber = 2
    userCodeRestriction = "\d{4}"
    value = false
}
```

Field	Type	Required	Description
event	string	+	type of an event
_id	string	+	an id of the item
deviceId	string	+	related device._id
deviceName	string	+	related device.name
deviceCategory	string	+	a device category
deviceSubcategory	string	+	a device subcategory
roomName	string	+	a room name

notifications	JsonAr ray	+	List of user ids for sending broadcast notification to users(null - all users / - no one)
name	string	+	see item.name(type)
deviceArmed	bool	-	related device.armed state
elementsMaxNumber	int	-	max allowed elements of a dictionary value
userCodeRestriction	string	-	restriction for a userCode code field http://www.lua.org/manual/5.3/manual.html#6.4.1
__ITEM_CHANGEABLE_FIELD	any	+	

Item fields which may be updated (__ITEM_CHANGEABLE_FIELD):

Field	Type	Description
show	bool	whether to show the item (on the UI) or not
valueFormatted	string	an item value formatted
value	object	an item value
valueMin	object	lower limit of item's value field
valueMax	object	upper limit of item's value field

item_dictionary_updated

sent when some changes happened to an dictionary item.

```
{
    event = "item_dictionary_updated"
    _id = "588b7eb528b12d03be86f36f"
    deviceId = "588t7eb528b12d03be86f36f"
    deviceName = "Touchscreen Deadbolt"
    deviceArmed = false
    deviceCategory = "door_lock"
    deviceSubcategory = ""
    roomName = ""
    notifications = { "12314324", "978343" }
    name = "user_codes"
    operation = "added"
    elementNumber = "1"
    value = {
        code = "Alex"
        name = "1234"
    }
}
```

Field	Type	Required	Description
event	string	+	type of an event
_id	string	+	an id of the item
deviceId	string	+	related device._id
deviceName	string	+	related device.name

deviceCategory	string	+	a device category
deviceSubcategory	string	+	a device subcategory
roomName	string	+	a room name
notifications	JsonArray y	+	List of user ids for sending broadcast notification to users(null - all users / - no one)
name	string	+	see item.name(type)
deviceArmed	bool	-	related device.armed state
elementsMaxNumber	int	+	max allowed elements of a dictionary value
userCodeRestriction	string	-	restriction for a userCode code field Lua Manual
operation	string	+	The operation field defines the initiator of this broadcast: hub.item.dictionary.value.add - `added` hub.item.dictionary.value.set - `updated` hub.item.dictionary.value.remove - `removed`
elementNumber	string	+	The element number.
value	any	+	The element value.

item_removed

sent when an item got unregistered from some device (can be sent only as a part of a device.removed sequence).

```
{
    event = "item_removed"
    _id = "588b7eb528b12d03be86f36f"
}
```

Field	Type	Required	Description
event	string	+	type of an event
_id	string	+	an id of the item