



How to **develop for Hue?**

Develop

Get Started

Application Design Guidance

Hue API

Hue Entertainment

Tools and SDKs

Overview

6. Rules API

Datatypes and Time Patterns

1. Lights API

2. Groups API

3. Schedules API

4. Scenes API

5. Sensors API

6. Rules API

7 Configuration API

8. Info API
(deprecated as of 1.15)

9. Resourcelinks API

10. Capabilities API

6.1. Get all rules

URL	<code>/api/<username>/rules</code>
Method	<code>GET</code>
Version	1.3
Permission	Whitelist

On this page:

[6.1. Get all rules](#)

[6.2. Get Rule](#)

[6.3. Create Rule](#)

[6.4. Update Rule](#)

[6.5. Delete Rule](#)

6.1.1. Description

Gets a list of all rules that are in the bridge.

6.1.2. Response

Returns a list of all rules in the system.

If there are no rules in the system then the bridge will return an empty object, {}.

6.1.3. Sample Response

```
{
  "1": {
    "name": "Wall Switch Rule",
    "lasttriggered": "2013-10-17T01:23:20",
    "creationtime": "2013-10-10T21:11:45",
    "timestriggered": 27,
    "owner": "78H56B12BA",
    "status": "enabled",
    "conditions": [
```

Remote API Quick start guide	{
	"address": "/sensors/2/state/buttonevent",
	"operator": "eq",
	"value": "16"
	},
Remote Authentication	{
	"address": "/sensors/2/state/lastupdated",
	"operator": "dx"
	}
Remote Hue API - Error Messages],
	"actions": [
	{
	"address": "/groups/0/action",
	"method": "PUT",
	"body": {
	"scene": "S3"
	}
Error messages	}
	}
Message Structure and Response]
	}
Supported Devices	"2": {
	"name": "Presence sensor rule",
	(...)
API Documentation Changelog	}}

6.2. Get Rule

Glossary terms

URL	<code>/api/<username>/rules/<id></code>
Method	<code>GET</code>
Version	1.3
Permission	Whitelist

6.2.1. Description

Returns a rule object with id matching <id> or an error if <id> is not available.

6.2.2. Sample Response

```
{
  "name": "Wall Switch Rule",
  "owner": "ruleOwner",
  "created": "2014-07-23T15:02:56",
  "lasttriggered": "none",
  "timestriggered": 0,
  "status": "enabled",
  "conditions": [
    {
      "address": "/sensors/2/state/buttonevent",
      "operator": "eq",
      "value": "16"
    },
    {
      "address": "/sensors/2/state/lastupdated",
```

```

        "operator": "dx"
      }
    ],
    "actions": [
      {
        "address": "/groups/0/action",
        "method": "PUT",
        "body": {
          "scene": "S3"
        }
      }
    ]
  }
}

```

6.3. Create Rule

URL	<code>/api/<username>/rules</code>
Method	<code>POST</code>
Version	1.3
Permission	Whitelist

6.3.1. Description

Creates a new rule in the bridge rule engine. A rule must contain at least 1 condition and 1 action and all conditions must evaluate to true for the action to be performed. Rules are deactivated if any errors are detected during evaluation. The number of rules, conditions and actions that are available on a bridge can be retrieved by reading the `/api/<username>/capabilities` endpoint.

6.3.2. Body

The below rule translates to:

"If sensor 2 has created an event with number 16 then activate scene S3.

```

{
  "name": "Wall Switch Rule",
  "conditions": [
    {
      "address": "/sensors/2/state/buttonevent",
      "operator": "eq",
      "value": "16"
    }
  ],
  "actions": [
    {
      "address": "/groups/0/action",
      "method": "PUT",
      "body": {
        "scene": "S3"
      }
    }
  ]
}

```

6.3.3. Sample Response

```

[[
  {
    "success": { "id": "3" }
  }
]]

```

6.3.4. Notes

The following attributes allowed in a condition:

address	string 1..32	Path to an attribute of a sensor resource. eq, gt, lt, dx or ddx (equals, greater than, less than, value, or delayed value has changed). ddx is introduced in 1.13
operator	string 2..2	stable , not stable (stable or not stable for a given time). Introduced in 1.13 in , not in (Current time is in or not in given time interval (only for /config/localtime, not UTC)). Introduced in 1.14
value	string 1..64	The resource attribute is compared to this value using the given operator. The value is cast to the data type of the resource attribute (in case of time, casted to a timePattern). If the cast fails or the operator does not support the data type the value is cast to the rule is rejected

The following attributes allowed in an action object:

address	string 1..32	Path to a light resource, a group resource or any other bridge resource
method	string 3..6	The HTTPS method used to send the body to the given address. Either "POST", "PUT", "DELETE" for local addresses.
body	string 1..90	JSON string to be send to the relevant resource.

The following operators are allowed:

Operator	Type	Example Usage
eq	equal	Used for bool and int.
dx,ddx	on change	Time (timestamps) int and bool values. Only dx or ddx is allowed, but not both. Triggers when value of button event is changed or change of presence is detected. ddx is introduced in 1.13
stable, not stable	on change	Time (timestamps) int and bool values. An attribute has or has not changed for a given time. Does not trigger a rule change. Not allowed on /config/utc and /config/localtime. Introduced in 1.13

Operator	Type	Example Usage
in, not in	on change	Current time is in or not in given time interval (only for /config/localtime, not UTC). "in" rule will be triggered on starttime and "not in" rule will be triggered on endtime. Only one "in" operator is allowed in a rule. Multiple "not in" operators are allowed in a rule. Not allowed to be combined with "not in". Introduced in 1.14
lt, gt	less than and greater than	Allowed on int values.

To create rules for the Hue Tap for example (e.g. If button 2 pressed then activate scene xxxxx) then the button event mapping is as follows:

Button	Event Code
Button 1	34
Button 2	16
Button 3	17
Button 4	18

The following errors can occur upon rule creation.

Code	Type	Description
601	RuleEngineFull	The Rule Engine has reached its maximum capacity of 100 rules.
607	ConditionError	Rule conditions contain errors or operator combination is not allowed
608	ActionError	Rule actions contain errors or multiple actions with the same resource address.
11		Too many items in the list (too many conditions or too many actions)

6.4. Update Rule

URL	<code>/api/<username>/rules/<id></code>
Method	<code>PUT</code>

URL	<code>/api/<username>/rules/<id></code>
Version	1.3
Permission	Whitelist

6.4.1. Description

Updates a rule in the bridge rule engine. At least one attribute has to be provided.

6.4.2. Body Example

```
{
  "actions": [{
    "address": "/groups/0/action",
    "method": "PUT",
    "body": {
      "scene": "S3"
    }
  }]
}
```

6.4.3. Sample Response

```
[
  {
    "success": {
      "/rules/1/actions": [
        {
          "address": "/groups/0/action",
          "method": "PUT",
          "body": {
            "scene": "S3"
          }
        }
      ]
    }
  }
]
```

6.5. Delete Rule

URL	<code>/api/<username>/rules/<id></code>
Method	<code>DELETE</code>
Version	1.3
Permission	Whitelist

6.5.1. Description

Deletes the specified rule from the bridge.

6.5.2. Response

The response details whether the rule was successfully removed from the bridge.

6.5.3. Sample Response

```
[{  
  "success": "/rules/1 deleted."  
}]
```



Connect with us



[Contact](#) [Terms & Conditions](#) [Privacy](#) [Product Security](#)

©2018-2019 Signify Holding. All rights reserved.