

Activate rules on input

Step 1. Find the rule id

After you created the rule to send an alarm for example like this:

Action

Do **Do HTTP request**

Interval of action : No more than once per **1** **sec**

HTTP URL **http://192.168.2.101:25123**

HTTP content **NBA001%647431eb-ca0d-a28f-ece6-9d9578319fea**

Content type **text/plain**

Login **Login to authenticate (optional)** Authentication type **Auto**

Password **Password to authenticate (opti...** Request type **Auto**

Log in to the Wave server web-interface, go to the tab for developers and then API testing tool click on the getEventRules and then test method:

getEventRules (X)

return all event rules.

GET: **/ec2/getEventRules**

format
Data format. Default value is "json".

id uuid
Event rule unique id. If omitted, return da

Använd inloggningsuppgifter
Lägg till inloggning och lösenord till url

Begärd URL **https://192.168.2.132:7001/ec2/getEver**

Testmetod

Status **200: OK**

Copy all text in the Answer box and past in for example Notepad++ (it will make it easier to search)

Then you can search for the text in the alarm rule you created earlier, in the example above I found this when searching

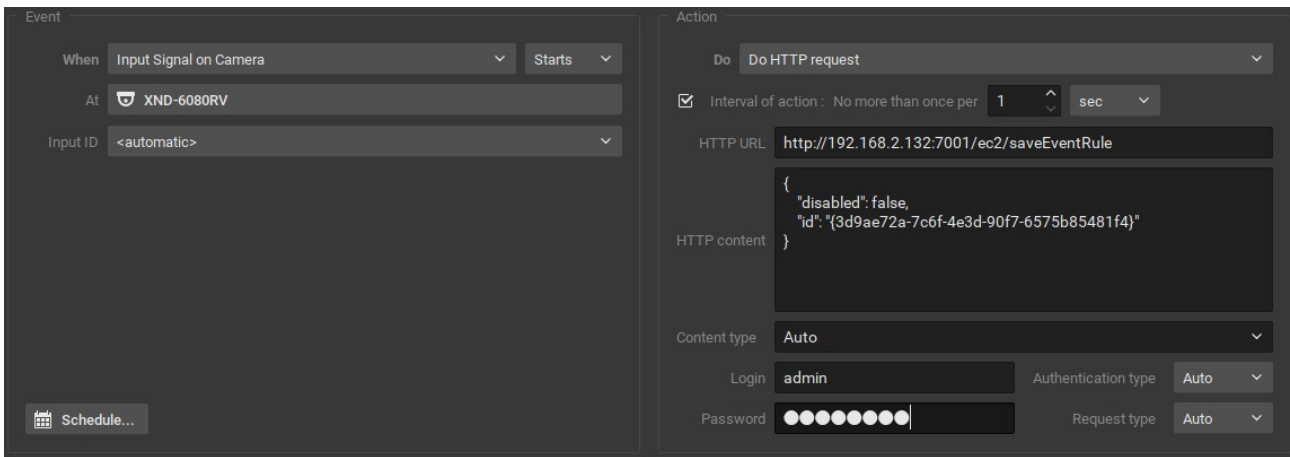
```
{
  "actionParams": {"allUsers":false,"authType":"authBasicAndDigest","contentType":"text/plain","durationMs":5000,"forced":t
  \recordBeforeMs":1000,"requestType":"","streamQuality":"highest","text":"NBA001%647431eb-ca0d-a28f-ec6-9d9578319fea","ur
  "actionResourceIds": [],
  "actionType": "execHttpRequestAction",
  "aggregationPeriod": 1,
  "comment": "",
  "disabled": false,
  "eventCondition": "{\"caption\":\"Alarm till kameran\",\"description\":\"_wipers_on\",\"eventTimestampUsecl\":\"0\",\"eventType\":\"undef
  {\"allUsers\":false,\"instigators\":[\"{00000000-0000-0000-0000-100000000000}\",\"level\":\"0\"},\"omitDbLogging\":false,\"reasonCode\"
  "eventResourceIds": [
    "{1e61d3fb-55d6-c8e4-04f3-23e07d2d0103}"
  ],
  "eventState": "Undefined",
  "eventType": "softwareTriggerEvent",
  "id": "{3d9ae72a-7c6f-4e3d-90f7-6575b85481f4}",
  "schedule": "",
  "system": false
},
```

The interesting part is id, and disabled so copy the lines and make the text look like this:

```
{
  "disabled": false,
  "id": "{3d9ae72a-7c6f-4e3d-90f7-6575b85481f4}"
}
```

Notice the comma after disabled and no comma after the id line

Then create a new rule with input as an event and http request as action:



The URL should be: <http://192.168.2.132:7001/ec2/saveEventRule>
change ip to fit your server setup.
Content is what you have earlier from the search:

To enable a rule:

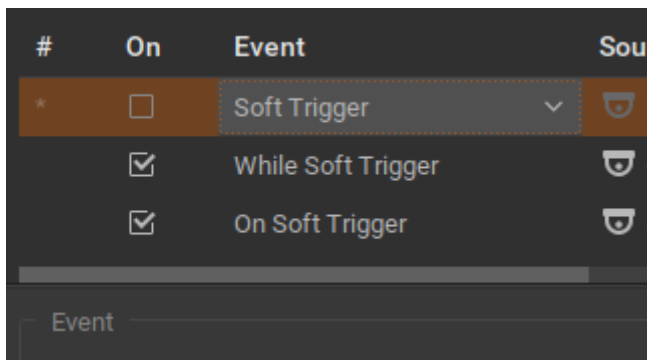
```
{  
  "disabled": false,  
  "id": "{3d9ae72a-7c6f-4e3d-90f7-6575b85481f4}"  
}
```

To disable a rule:

```
{  
  "disabled": true,  
  "id": "{3d9ae72a-7c6f-4e3d-90f7-6575b85481f4}"  
}
```

Create several rules of input starts and stop event to activate or deactivate rules.

To see that it is working you can look at the list of rules and when you send the command to disable a rule you will see that the checkbox is unchecked for that specific rule:



This approach can also be used for other things in the API like turn on or off recording of a camera:

For example, send this to /ec2/saveCameraUserAttributes:

```
{  
  "scheduleEnabled": true,  
  "id": "{e8f65f73-6773-8f87-3676-59f7cb169fcb}"  
}
```