

system.tag.editAlarmConfig

This function is used in **Python Scripting**.

Description

Edit the alarm configuration of multiple existing tags in Ignition with a single call.

Client Permission Restrictions

Permission Type: Tag Editing

Client access to this scripting function is blocked to users that do not meet the role/zone requirements for the above permission type. This function is unaffected when run in the Gateway scope.

Syntax

system.tag.editAlarmConfig(tagPaths, alarmConfig)

- Parameters

String[] tagPaths - The full path to the tag you want to edit. Note: you can specify the tag provider name in square brackets at the beginning of the parentPath string. Example: "[myTagProvider]MyTagsFolder". If the tag provider name is left off then the project default provider will be used.

PyDictionary alarmConfig - A dictionary of multi-dimensional lists containing the new alarm configuration. The key in the dictionary will be the name of the alarm being to edit, and the value is a list of lists. The nested lists use the format ["name", "Value", "newValue"]. Note that item 1 is always "Value".

Example: {"targetAlarm": [{"propertyToChange", "Value", "newValue"}, {"anotherProperty", "Value", "anotherNewValue"}]}

If the name of the alarm does not match a pre-existing alarm, **a new alarm will be created**. Note that alarm names are case sensitive. Beware of typos.

A list of scripting names for each alarm property can be found on the [Tag Properties](#) page.

- Returns

nothing

- Scope

All

Code Examples

Code Snippet - Single Tag, Multiple properties

```
# The following example will alter the alarm configuration on a
single tag.
# The tag currently has an alarm named "High Temp". The code
below will change the name of the alarm
# to "Low Temp", and change the Setpoint to 100.

# Build a list of tag paths. Only a single tag will be altered,
so create a list of one item and store the list in a variable
tagPaths = ["sensors/B3:0"]

# Build the dictionary of alarm names, and properties to change.

alarmConfig = {"High Temp": [{"name", "Value", "Low
Temp"}, {"setpointA", "Value", "100"}]}

# Edit the alarm configuration.
system.tag.editAlarmConfig(tagPaths, alarmConfig)
```

Code Snippet - Multiple Tags, Single Alarm per Tag

```
# The following example will disable alarms on multiple tags.
# The tags below both have an alarm named "Alarm"
#This code will edit the Enabled property for both alarms.

# Build a list of tag paths.
tagPaths = ["Tanks/Tank_1/level_PV", "Tanks/Tank_2/level_PV"]

# Build a dictionary of alarms. Both alarms have the name
"Alarm", so a dictionary of one item
# will be able to alter both alarms in a single call.
alarmConfig = {"Alarm": [{"enabled", "Value", "0"}]}

# Edit the alarm configuration.
system.tag.editAlarmConfig(tagPaths, alarmConfig)
```

Code Snippet - Multiple Tags, Multiple Alarms per Tag

```
# The following example will create two alarms each on two
different tags.
# The code assumes there are not pre-existing alarms on the tags
by the name of "High Level" and "Low Level"
# The Name, Mode, and Setpoint properties will be modified for
each alarm.

# Build a list of tag paths.
tagPaths = ["Tanks/Tank_1/level_PV", "Tanks/Tank_2/level_PV"]

# Configure two alarms on the tags.
# Mode value of 2 = Above Setpoint
# Mode value of 3 = Below Setpoint
alarmConfig = {"High
Level": [{"mode", "Value", "2"}, {"setpointA", "Value", 80}], "Low
Level": [{"mode", "Value", "3"}, {"setpointA", "Value", 15}]}

# Edit the alarm configuration.
system.tag.editAlarmConfig(tagPaths, alarmConfig)
```