system.alert.queryAlertHistory

Ignition version 7.6 introduced a new alarming system. All of the associated functions can be found under the system.alarm section.

Description

This function queries one of the configured Alert Storage profiles for alert history. The filter arguments help to narrow down the results to alerts that match various criteria, most commonly a range of dates. You can use * to match any number of characters and ? to match a single character in the filter string arguments.

System - The system that issued the alert.

Path - The path to the alert item

Display Path - The custom display path (if any) for the alert. Will be the Path if no Display Path is configured.

State Name - The state name for the alert.

Severity - The severity, as a string.

Severity Code - The severity as an integer. 0-4, low-high.

Active - A boolean indicating whether this alert state is still active.

Active Timestamp - The time at which this alert went active.

Active Value - The value that triggered this alert to go active.

Cleared - A boolean indicating whether this alert has cleared.

Cleared Timestamp - The time at which this alert cleared. May be null.

Cleared Value - The value that cleared the alert.

Acked - A boolean indicating whether or not this alert was been acknowledged.

Ack Timestamp - The time that the alert was acknowledged. May be null.

Ack user - The user who acknowledged the alert.

Notes - The notes field for the alert

Flags - A bitmask representing the current alert state. 0x01= Active, 0x02=Cleared, 0x04=Acknowledged. So if the alert is active and acknowledged, but not cleared, this will be 0x01 | 0x04 = 5

Syntax

system.alert.queryAlertHistory(storageProfile, startDate, endDate, system, path, stateName, minSeverity, maxSeverity, activeAndUnacked, activeAndAcked, clearAndUnacked, clearAndAcked, sortOrder, displayPath)

- Parameters

  String storageProfile - The name of the alert storage profile to query.

  Date startDate - Earliest alert to return. Defaults to 8 hours earlier than current time if omitted.

  Date endDate - Latest alert to return. Defaults to current time if omitted.

  String system - Filter string to restrict results based on the alert system.

  String path - Filter string to restrict results based on the alert path.

  String stateName - Filter string to restrict results based on the alert state name.

  Integer minSeverity - Minimum severity to return. Defaults to 0 (Low).

  Integer maxSeverity - Maximum severity to return. Defaults to 4 (High).

  Boolean activeAndUnacked - Whether or not to return alerts that are currently active and unacknowledged. Default is true.

  Boolean activeAndAcked - Whether or not to return alerts that are currently active and have been acknowledged. Default is true.

  Boolean clearAndUnacked - Whether or not to return alerts that are cleared and unacknowledged. Default is true.

  Boolean clearAndAcked - Whether or not to return alerts that are cleared and have been acknowledged. Default is true.

  String sortOrder - The sort order in which to return matching alerts. Either "asc" or "desc", referring to the alert's active timestamp. Default is "desc".

  String displayPath - Filter string to restrict results based on the alert's display path.

- Returns

  Dataset - A dataset containing the historical alert events from the given storage profile that matched the filter and date range arguments.

- Scope

  All

Code Examples

Code Snippet

```
 #code would query an alert storage profile called "DBHistory", looking for the number of unacknowledged
alerts in the last 36 hours, displaying the number to the user in a popup message.

from java.util import Date
from java.util import Calendar

cal = Calendar.getInstance()

end = cal.getTime()
cal.add(Calendar.HOUR, -36)
start = cal.getTime()

results = system.alert.queryAlertHistory("DBHistory", start, end,
 activeAndAcked=0, clearAndAcked=0)

if results.rowCount > 0:
   system.gui.messageBox("There are %d un-acked alerts!" % results.rowCount)
```