1 Trigger-based event correlation

Overview

Trigger-based event correlation allows to correlate separate problems reported by one trigger.

While generally an OK event can close all problem events created by one trigger, there are cases when a more detailed approach is needed. For example, when monitoring log files you may want to discover certain problems in a log file and close them individually rather than all together.

This is the case with triggers that have **Multiple Problem Event Generation** enabled. Such triggers are normally used for log monitoring, trap processing, etc.

It is possible in Zabbix to relate problem events based on the event tags. Tags are used to extract values and create identification for problem events. Taking advantage of that, problems can also be closed individually based on matching tag.

In other words, the same trigger can create separate events identified by the event tag. Therefore problem events can be identified one-by-one and closed separately based on the identification by the event tag.

**How it works**

In log monitoring you may encounter lines similar to these:

```
Line1: Application 1 stopped
Line2: Application 2 stopped
Line3: Application 1 was restarted
Line4: Application 2 was restarted
```

The idea of event correlation is to be able to match the problem event from Line1 to the resolution from Line3 and the problem event from Line2 to the resolution from Line4, and close these problems one by one:

```
Line1: Application 1 stopped
Line3: Application 1 was restarted #problem from Line 1 closed

Line2: Application 2 stopped
Line4: Application 2 was restarted #problem from Line 2 closed
```

To do this you need to tag these related events as, for example, “Application 1” and “Application 2”. That can be done by applying a regular expression to the log line to extract the tag value. Then, when events are created, they are tagged “Application 1” and “Application 2” respectively and problem can be matched to the resolution.

**Configuration**
To configure event correlation on trigger level:

- go to the trigger configuration form

- select 'Problem event generation mode' as *Multiple*
- select that 'OK event closes' *All problems if tag values match*
- enter the name of the tag for event matching
- configure the tags to extract tag values from log lines

If configured successfully you will be able to see problem events tagged by application and matched to their resolution in *Monitoring → Problems*.

Because misconfiguration is possible, when similar event tags may be created for unrelated problems, please review the cases outlined below!

- With two applications writing error and recovery messages to the same log file a user may decide to use two *Application* tags in the same trigger with different tag values by using separate regular expressions in the tag values to extract the names of, say, application A and application B from the *ITEM.VALUE* macro (e.g. when the message formats differ). However, this may not work as planned if there is no match to the regular expressions. Non-matching regexps will yield empty tag values and a single empty tag value in both problem and OK events is enough to correlate them. So a recovery message from application A may accidentally
close an error message from application B.

- Actual tags and tag values only become visible when a trigger fires. If the regular expression used is invalid, it is silently replaced with an *UNKNOWN* string. If the initial problem event with an *UNKNOWN* tag value is missed, there may appear subsequent OK events with the same *UNKNOWN* tag value that may close problem events which they shouldn't have closed.

- If a user uses the {ITEM.VALUE} macro without macro functions as the tag value, the 255-character limitation applies. When log messages are long and the first 255 characters are non-specific, this may also result in similar event tags for unrelated problems.