

1 宏函数

概述

宏函数能提供自定义宏值的功能。

有时候宏可能会解析为一个不一定易于使用的值。它可能很长，或包含你想提取的一个特殊感兴趣的子字符串。这在宏函数中是可以使用的。

宏函数的语法为：

```
{<macro>.<func>( <params> )}
```

其中：

- <macro> - 这个参数为要定义的宏（例如 {ITEM.VALUE}[]
- <func> - 要应用的函数；
- <params> - 以逗号分隔的函数参数列表。如果他们以（空格），" 或者包含), , 这些符号开始，则参数必须要引用。

例如：

```
{{ITEM.VALUE}.regsub(pattern, output)}
```

受支持的宏函数

函数	描述	参数	受支持于
regsub (<pattern>,<output>)			
	通过正则表达式匹配提取的子字符串（区分大小写）。	<p>pattern - 匹配的正则表达式</p> <p>output - 输出的选项。 \1 - \9 占位符支持被正则表达式匹配的组。placeholders are supported for captured groups.</p> <p>如果参数 pattern 是一个不正确的正则表达式，那么将返回 "UNKNOWN" []</p>	{ITEM.VALUE} {ITEM.LASTVALUE}
iregsub (<pattern>,<output>)			
	通过正则表达式匹配提取的子字符串（区分大小写）。	<p>pattern - 匹配的正则表达式</p> <p>output - 输出得选项 \1 - \9 placeholders are supported for captured groups</p> <p>如果参数 pattern 是一个不正确的正则表达式，那么将返回 "UNKNOWN" []</p>	{ITEM.VALUE} {ITEM.LASTVALUE}

如果在受支持的位置使用函数，但是应用于不支持宏函数得宏，那么宏的计算结果为 "UNKNOWN"[]

如果不支持宏函数的位置将宏函数应用于宏，则忽略该函数。

示例

关于宏函数可用于自定义宏值的方法，在下面的示例中说明，其中包含的“log line”作为接收值：

接收值	宏	输出
123Log line	<code>{{ITEM.VALUE}.regsub(^[0-9]+, Problem)}</code>	Problem
123 Log line	<code>{{ITEM.VALUE}.regsub("^[0-9]+", "Problem")}</code>	Problem
123 Log line	<code>{{ITEM.VALUE}.regsub("^[0-9]+", Problem ID: \1)}</code>	Problem ID: 123
Log line	<code>{{ITEM.VALUE}.regsub(".*", "Problem ID: \1")}</code>	Problem ID:
MySQL crashed errno 123	<code>{{ITEM.VALUE}.regsub("^[A-Z]+.*([0-9]+)", "Problem ID: \1_\2 ")}</code>	Problem ID: MySQL_123?
123 Log line	<code>{{ITEM.VALUE}.regsub("([1-9]+", "Problem ID: \1")}</code>	*UNKNOWN* (invalid regular expression)

From: <https://www.zabbix.com/documentation/3.4/> - Zabbix Documentation 3.4

Permanent link: https://www.zabbix.com/documentation/3.4/zh/manual/config/macros/macro_functions

Last update: 2017/08/22 08:19

