

1 Creating an item

Overview

To create an item in Zabbix frontend, do the following:

- Go to: *Configuration* → *Hosts*
- Click on *Items* in the row of the host
- Click on *Create item* in the upper right corner of the screen
- Enter parameters of the item in the form

Configuration

Item : Available memory

Host: Zabbix server [Select]

Name: Available memory

Type: Zabbix agent

Key: vm.memory.size[available] [Select]

Host interface: 192.168.3.41 : 10050

Type of information: Numeric (unsigned)

Data type: Decimal

Units: B

Use custom multiplier: [1]

Update interval (in sec): 60

Flexible intervals

Interval	Period	Action
No flexible intervals defined.		

New flexible interval: Interval (in sec) [50] Period [1-7,00:00-24:00] [Add]

Keep history (in days): 7

Keep trends (in days): 365

Store value: As is

Show value: As is [show value mappings]

New application: []

Applications: -None-, CPU, Filesystems, General, Memory, Network interfaces

Populates host inventory field: -None-

Description: Available memory is defined as free+cached+buffers memory.

Status: Enabled

[Save] [Cancel]

Item attributes:

Parameter	Description
<i>Host</i>	Select the host or template.
<i>Name</i>	This is how the item will be named. The following macros can be used: \$1, \$2...\$9 - referring to the first, second... ninth parameter of the item key For example: Free disk space on \$1 If the item key is "vfs.fs.size[/,free]", the description will automatically change to "Free disk space on /"
<i>Type</i>	Item type. See individual item type sections.
<i>Key</i>	Item key. The supported item keys can be found in individual item type sections. The key must be unique within a single host. If key type is 'Zabbix agent', 'Zabbix agent (active)', 'Simple check' or 'Zabbix aggregate', the key value must be supported by Zabbix agent or Zabbix server. See also: the correct key format .
<i>Host interface</i>	Select the host interface. This field is available when editing an item on the host level.
<i>Type of information</i>	Type of data as stored in the database after performing conversions, if any. Numeric (unsigned) - 64bit unsigned integer Numeric (float) - floating point number Negative values can be stored. Allowed range (for MySQL): -999999999999.9999 to 999999999999.9999 (double(16,4)). Character - character (string) data limited to 255 bytes Log - log file. Must be set for log*, eventlog item keys. Text - text of unlimited size
<i>Data type</i>	Data type is used for integer items in order to specify the expected data type: Boolean - textual representation translated into either 0 or 1. Thus, 'TRUE' is stored as 1 and 'FALSE' is stored as 0. All values are matched in a case-insensitive way. Currently recognized values are, for: <i>TRUE</i> - true, t, yes, y, on, up, running, enabled, available <i>FALSE</i> - false, f, no, n, off, down, unused, disabled, unavailable Additionally, any non-zero numeric value is considered to be TRUE and zero is considered to be FALSE. Octal - data in octal format Decimal - data in decimal format Hexadecimal - data in hexadecimal format Zabbix will automatically perform the conversion to numeric. The conversion is done by Zabbix server (even when a host is monitored by Zabbix proxy).

Parameter	Description
<i>Units</i>	<p>If a unit symbol is set, Zabbix will add post processing to the received value and display it with the set unit postfix.</p> <p>By default, if the raw value exceeds 1000, it is divided by 1000 and displayed accordingly. For example, if you set <i>bps</i> and receive a value of 881764, it will be displayed as 881.76 Kbps.</p> <p>Special processing is used for B (byte), Bps (bytes per second) units, which are divided by 1024. Thus, if units are set to B or Bps Zabbix will display:</p> <p>1 as 1B/1Bps 1024 as 1KB/1KBps 1536 as 1.5KB/1.5KBps</p> <p>Special processing is used if the following time-related units are used:</p> <p>unixtime - translated to "yyyy.mm.dd hh:mm:ss". To translate correctly, the received value must be a <i>Numeric (unsigned)</i> type of information.</p> <p>uptime - translated to "hh:mm:ss" or "N days, hh:mm:ss" For example, if you receive the value as 881764 (seconds), it will be displayed as "10 days, 04:56:04"</p> <p>s - translated to "yyy mmm ddd hhh mmm sss ms"; parameter is treated as number of seconds. For example, if you receive the value as 881764 (seconds), it will be displayed as "10d 4h 56m" Only 3 upper major units are shown, like "1m 15d 5h" or "2h 4m 46s". If there are no days to display, only two levels are displayed - "1m 5h" (no minutes, seconds or milliseconds are shown). Will be translated to "< 1 ms" if the value is less than 0.001. See also the unit blacklist.</p>
<i>Use custom multiplier</i>	<p>If you enable this option, all received values will be multiplied by the integer or floating-point value set in the value field.</p> <p>Use this option to convert values received in KB, MBps, etc into B, Bps. Otherwise Zabbix cannot correctly set prefixes (K, M, G etc).</p>
<i>Update interval (in sec)</i>	<p>Retrieve a new value for this item every N seconds.</p> <p><i>Note:</i> If set to '0', the item will not be polled. However, if a flexible interval also exists with a non-zero value, the item will be polled during the flexible interval duration.</p>

Parameter	Description
Flexible intervals	<p>You can create exceptions to <i>Update interval</i>. For example: Interval: 10, Period: 1-5,09:00-18:00 - will check the item every 10 seconds during working hours. Interval: 0, Period: 1-7,00:00-7:00 - will disable checking the item at night. Interval: 0, Period: 7-7,00:00-24:00 - will disable checking the item on Sundays. To check an item once per day at a specific time (say, 12:00), set the default <i>Update interval</i> to '0', but specify 60 in the flexible interval and a period like 1-7,12:00-12:01</p> <p>Up to seven flexible intervals can be defined. If multiple flexible intervals overlap, the smallest <i>Interval</i> value is used for the overlapping period. Note that if the smallest value of overlapping flexible intervals is '0', no polling will take place. Outside the flexible intervals the default update interval is used.</p> <p>See the page about setting time periods for description of the <i>Period</i> format. Note that if the flexible interval equals the length of the period, the item will be checked exactly once. If the flexible interval is greater than the period, the item might be checked once or it might not be checked at all (thus such configuration is not advisable). If the flexible interval is less than the period, the item will be checked at least once.</p> <p>If the flexible interval is set to '0', the item is not polled during the flexible interval period and resumes polling according to the default <i>Update interval</i> once the period is over.</p> <p><i>Note:</i> Not available for Zabbix agent active items.</p>
Keep history (in days)	<p>Number of days to keep detailed history in the database. Older data will be removed by the housekeeper.</p> <p>It is recommended to keep the recorded values for the smallest possible number of days to reduce the size of value history in the database. Instead of keeping long history of values, you can keep longer data of trends.</p> <p>See also History and trends.</p>
Keep trends (in days)	<p>Keep aggregated (hourly min, max, avg, count) detailed history for N days in the database. Older data will be removed by the housekeeper.</p> <p><i>Note:</i> Keeping trends is not available for non-numeric data - character, log and text.</p> <p>See also History and trends.</p>

Parameter	Description
<i>Store value</i>	<p>As is - no pre-processing</p> <p>Delta (speed per second) - evaluate value as $(value - prev_value) / (time - prev_time)$, where <i>value</i> - current value <i>value_prev</i> - previously received value <i>time</i> - current timestamp <i>prev_time</i> - timestamp of previous value</p> <p>This setting is extremely useful to get speed per second for a constantly growing value.</p> <p>If current value is smaller than the previous value, Zabbix discards that difference (stores nothing) and waits for another value. This helps to work correctly with, for instance, a wrapping (overflow) of 32-bit SNMP counters.</p> <p><i>Note:</i> As this calculation may produce floating point numbers, it is recommended to set the 'Type of information' to <i>Numeric (float)</i>, even if the incoming raw values are integers. This is especially relevant for small numbers where the decimal part matters. If the floating point values are large and may exceed the 'float' field length in which case the entire value may be lost, it is actually suggested to use <i>Numeric (unsigned)</i> and thus trim only the decimal part.</p> <p>Delta (simple change) - evaluate as $(value - prev_value)$, where <i>value</i> - current value <i>value_prev</i> - previously received value</p> <p>This setting can be useful to measure a constantly growing value. If the current value is smaller than the previous value, Zabbix discards that difference (stores nothing) and waits for another value.</p>
<i>Show value</i>	<p>Apply value mapping to this item. Value mapping does not change received values, it is for displaying data only.</p> <p>It works with integer items only.</p> <p>For example, "Windows service states".</p>
<i>Log time format</i>	<p>Available for items of type Log only. Supported placeholders:</p> <ul style="list-style-type: none"> * y: Year (1970-2038) * M: Month (01-12) * d: Day (01-31) * h: Hour (00-23) * m: Minute (00-59) * s: Second (00-59) <p>If left blank the timestamp will not be parsed.</p> <p>For example, consider the following line from the Zabbix agent log file: " 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211)."</p> <p>It begins with six character positions for PID, followed by date, time, and the rest of the line.</p> <p>Log time format for this line would be "pppppp:yyyyMMdd:hhmmss".</p> <p>Note that "p" and ":" chars are just placeholders and can be anything but "yMdhms".</p>
<i>New application</i>	Enter the name of a new application for the item.
<i>Applications</i>	Link item to one or more existing applications.
<i>Populates host inventory field</i>	You can select a host inventory field that the value of item will populate. This will work if automatic inventory population is enabled for the host.
<i>Description</i>	Enter an item description.

Parameter	Description
Status	Enabled - the item will be processed. Disabled - the item will not be processed. Not supported - the item is not supported. This item will not be processed, however Zabbix may try to periodically set the status of such items to <i>Enabled</i> according to the interval set for refreshing unsupported items .

You can also create an item by opening an existing one, pressing the *Clone* button and then saving under a different name.

When editing an existing [template](#) level item on a host level, a number of fields are read-only. You can use the link in the form header and go to the template level and edit them there, keeping in mind that the changes on a template level will change the item for all hosts that the template is linked to.

Unit blacklist

By default, specifying a unit for an item will result in a multiplier prefix being added - for example, value 2048 with unit B would be displayed as 2KB. For a pre-defined, hardcoded list of units this is prevented:

- ms
- RPM
- rpm
- %

Note that both lowercase and uppercase **rpm** (*rpm* and *RPM*) strings are blacklisted.

Unsupported items

An item can become unsupported if its value cannot be retrieved for some reason. Such items are still rechecked at a fixed interval, configurable in [Administration section](#).

From:

<https://www.zabbix.com/documentation/2.0/> - **Zabbix Documentation 2.0**

Permanent link:

<https://www.zabbix.com/documentation/2.0/manual/config/items/item>

Last update: **2018/06/29 08:13**

