

## 10 Upgrade notes for 3.4.0

These notes are for upgrading from Zabbix 3.2.x to Zabbix 3.4.0. All notes are grouped into:

- **Critical** - the most critical information related to the upgrade process and the changes in Zabbix functionality
- **Informational** - all remaining information describing the changes in Zabbix functionality

It is possible to upgrade to Zabbix 3.4.0 from versions before Zabbix 3.2.0. See the [upgrade procedure](#) section for all relevant information about upgrading from previous Zabbix versions.

### Critical

#### Dropping SQLite support for server/frontend

SQLite is no longer supported as a backend database for Zabbix server and frontend. It is still supported for Zabbix proxy.

#### Dropping IE9 and IE10 support

Support for Microsoft Internet Explorer 9 and Explorer 10 will not be provided any more.

#### Clearing web browser cookies/cache

After the upgrade you may need to clear web browser cookies and web browser cache for the frontend to work properly.

### Additional dependencies

- **libpcre** - the PCRE library is now mandatory when compiling Zabbix server. It is required for [Perl Compatible Regular Expression](#) (PCRE) support. The naming may differ depending on the GNU/Linux distribution, for example 'libpcre3' or 'libpcre1'. Note that you need exactly PCRE (v8.x); PCRE2 (v10.x) library is not used;
- **libevent** (version 1.4 or later) library is now mandatory:
  - for Zabbix server always;
  - for Zabbix proxy if compiling with IPMI monitoring.

### Case-sensitive MySQL database

A case-sensitive MySQL database is required for proper server work. It is [recommended](#) to create a case-sensitive MySQL database during new installations. If you created a MySQL database with the utf8 character set previously, in order to support case sensitiveness of stored data, you need to convert the charset to utf8\_bin.

## PCRE library for regular expressions

Regular expression support in Zabbix has been switched from POSIX extended regular expressions to [Perl Compatible Regular Expressions](#) (PCRE) for enhanced regular expressions and consistency with the frontend. While PCRE is mostly compatible with POSIX, still there are some differences such as that POSIX functions find the longest of the leftmost match, but PCRE stops on the first valid match. POSIX equivalence classes and collating elements are not supported.

As some regular expressions may yield different results after the upgrade, before upgrading you must review the regular expressions in use in supported [locations](#). Expressions where the previous syntax may lead to a different result after the upgrade must be manually converted to PCRE regular expression syntax yielding the same result as before the upgrade.

See also: [Differences between](#) POSIX extended and PCRE regular expressions.

GNU grep utility supports both POSIX extended regular expressions (with '-E' switch) and PCRE regular expressions (with '-P' switch). You can run it first with an '-E', then with a '-P' switch with the regular expression and analyze the test file. If both outputs are not identical to the expected, the regular expression should be modified to work correctly after upgrade.

Note that if you upgrade only server and proxies, but some hosts continue to use older Zabbix agents (before v3.4), those older agents still “understand” POSIX regular expressions only. Regular expressions used in their log[], logrt[] etc items should not be modified to PCRE syntax until the agents are upgraded.

## Command/script execution changes

Due to command/script exit code check [introduction](#) in Zabbix 3.4, [alertsscripts](#) can be executed multiple times if their exit code is different from 0. Previously configured items with user parameters executed by Zabbix server, external check items, and **system.run** items whose exit code is not 0 may become “Not supported” due to additional checks for exit code; behaviour of the items with “nowait” flag is not changed though.

## Configuration parameters

The SenderFrequency [configuration](#) parameter has been removed. It must be removed from server configuration for 3.4 when upgrading, otherwise you will get an error and the server **will not** start. Retry frequency of failed alerts is now looked up in the [media type](#) configuration (*Attempt interval* parameter) and is 10 seconds by default. The default of SenderFrequency used to be 30 seconds.

## URL-encoding in web scenarios

While [automatic URL-encoding](#) is supported starting with Zabbix 3.4, values in existing scenarios will not be automatically converted to URL-encoding. If you want those values URL-encoded you have to edit the [scenarios](#) manually using the new fields:

- move query variables from the URL into *Query fields*
- move post variables from *Raw data* into *Form data*

If you have already manually URL-encoded values they will be left as is and will not be double URL-encoded.

## Upgrading Java gateway

Zabbix Java gateway has to be upgraded to support the new functionality of:

- [configurable JMX endpoints](#);
- [low-level discovery of JMX objects](#).

## JMX item protocol changes

An example of the new protocol:

```
{"request": "java gateway  
jmx", "jmx_endpoint": "service:jmx:rmi:///jndi/rmi://127.0.0.1:12345/jmxrmi", "  
keys": ["jmx[\"java.lang:type={#FSNAME}\", \"HeapMemoryUsage.used\"]"]}
```

## Informational

### Configuration parameters

A new configuration parameter `SocketDir` has been added to Zabbix server and proxy configurations. This parameter points to the directory where internal Zabbix socket files are stored (/tmp by default). Server and proxy use different socket file names so it's safe to use the same `SocketDir` for server and proxy running on the same system. However, running multiple servers or proxies in the same system will require different `SocketDir` configurations.

### Dashboard changes

The dashboard section has seen a major [overhaul](#) in Zabbix 3.4, which brings a number of related changes:

- The *Last 20 issues* widget has been removed and replaced by a *Problems* widget, which is similar in layout to the *Monitoring → Problems* section. Because of this, the `DEFAULT_LATEST_ISSUES_CNT` frontend [definition](#) is not used any more and has been removed.
- *Problems* and *System Status* widgets no longer display triggers in the PROBLEM state if there is no corresponding event (i.e., the event is deleted)

## Map permission changes

Maps are no longer hidden if the user does not have read permission to all map elements. Instead, the map is shown, proving the user has read access to at least one map element. The elements that the user does not have read permission to are displayed with a grey icon and without any textual information on it (except for triggers which do have labels).

See also: [Network map configuration](#)

## Using templates for network devices

If you are importing the new out-of-the-box [templates for network devices](#), you may want to also update the @Network interfaces for discovery global regular expression to:

```
Result is FALSE: ^Software Loopback Interface
Result is FALSE: ^(In)?[lL]oop[bB]ack[0-9._]*$
Result is FALSE: ^NULL[0-9.]*$
Result is FALSE: ^[lL]o[0-9.]*$
Result is FALSE: ^[sS]ystem$
Result is FALSE: ^Nu[0-9.]*$
```

to filter out loopbacks and null interfaces on most systems.

## Support of macros and time suffixes in time periods

To allow for an extended macro and time suffix [support](#), several fields that previously stored their value in days have been converted to store the values **in seconds**, e.g.:

- History/trend storage period in items and item prototypes
- *Keep lost resources* setting in low-level discovery
- Global housekeeping settings

While the numeric values are adjusted automatically during an upgrade, it is a **different** case if the *Keep lost resources* setting contained a macro before the upgrade. Its value (in days) is no longer valid in this field and cannot be adjusted by simply appending 'd' to the macro, because that is not an allowed macro syntax. The macro value cannot be changed either, because the macro may be used in other places. Therefore in this case during an upgrade the *Keep lost resources* setting is automatically converted to the maximum value of 25 years. This is something to take care of manually after the upgrade.

## Parallel processing of alerts

Previously all problem notifications were handled by one alerter process. In the new version, **parallel**

**processing** of alerts is implemented. There is a new alert manager process that can distribute alerts to several “worker” processes. Media types are processed in parallel. The maximum number of concurrent sessions is configurable per media type, but the total number of alerter processes on server can only be limited by the new StartAlerters [parameter](#).

During an upgrade, the number of concurrent sessions is set as '1' for each media type, while the default value of StartAlerters is '3'. For the e-mail media type, you may want to [manually](#) increase the number of concurrent sessions or set it to '0' (unlimited) for faster processing of alerts.

Alert manager will now also fail alert scripts without access rights, without retrying.

## Database watchdog removed

The database watchdog process (db watchdog) has been removed and its tasks are now handled by the [alerter process](#). If you are monitoring your Zabbix server using the *Template App Zabbix Server* template you might get an unsupported item 'Zabbix busy db watchdog processes, in %'. In this case just remove or disable this item from the template.

## IT services renamed to services

Configuration and monitoring sections that previously were called “IT services” have been renamed to “Services” to reflect the fact that a broader range of services can be measured, not only IT.

## Item value preprocessing options

Item value preprocessing options have been [unified](#) and expanded as a separate tab in item configuration. As a result, several of the previously separate preprocessing parameters in items/item prototypes have been dropped:

- *Data type*, *Use custom multiplier* and *Store value* fields
- *data\_type*, *multiplier* and *delta* fields in API
- *data\_type*, *multiplier* and *delta* tags in XML export

During an upgrade, all of these properties in existing items/item prototypes are converted to the new preprocessing options automatically. Similarly, when importing XMLs from older versions, these options are converted to the new preprocessing options.

Additionally, *Delta (simple change)* and *Delta (speed per second)* preprocessing options have been renamed to *Simple change* and *Change per second*.

Too many data gathering processes (pollers, unreachable pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) together with IPMI manager, SNMP trapper and preprocessing workers can exhaust the per-process file descriptor limit for the preprocessing manager. This will cause Zabbix server to stop (usually shortly after the start, but sometimes it can take more time). The configuration file should be revised or the limit should be raised to avoid this situation.

## Item changes

Previously the *sleep* parameter of the **proc.num** agent item on FreeBSD, OpenBSD and NetBSD platforms incorrectly counted processes in both interruptible sleep and uninterruptible sleep state. Now it has been fixed to count processes only in interruptible sleep while the new *disk* parameter can be used to obtain process count in uninterruptible sleep.

## Macro changes

Since event correlation was introduced in Zabbix 3.2.0, the {TRIGGER.STATUS} macro could no longer be relied upon to accurately show problem status - the problem could already be resolved with the original trigger still in 'Problem' status. Therefore {TRIGGER.STATUS} has now been removed from default notification messages.

Instead, accurate problem status can now be acquired by using {EVENT.VALUE} and {EVENT.STATUS} macros. These macros have been changed and now return (for trigger and internal events):

- 'Problem' value if the original event is still in problem state;
- 'OK' value if the original event has been resolved.

## Syntax for selecting nested host groups

If upgrading from Zabbix 3.2.0 and 3.2.1, note that the syntax for including nested subgroups has changed.

In Zabbix 3.2.0 and 3.2.1 nested host groups are included with the parent host group, if the parent group is specified as `hostgroup/*`. Since [Zabbix 3.2.2](#) and in Zabbix 3.4.0, the `/*` syntax is dropped. Instead, nested host groups are included if simply the parent host group is specified as is. This means that a host group that is set, for example, in action conditions, now **silently** includes all its nested host groups.

## Protocol changes

Communication protocol between the frontend side and server side has been changed. A new "sid" parameter has been added to the request from frontend to server for script execution. This parameter takes the ID of session (authentication token) and it is used to check the user permissions to execute scripts. Also, an additional check for host permissions (READ/WRITE, READ and DENY) in the script has been added to this request. If the necessary access rights to the host are missing, the server will return a response that access to execute the script is denied.

## API changes

- Deprecated `isreadable()` and `iswritable()` methods as well as the `proxy.interfaces`

parameter have been removed.

- Validation of “period” field has been added disallowing a trailing semicolon. All existing trailing semicolons will be removed by proxy/server during upgrade.

## Changed housekeeper defaults

The default data storage period for internal, discovery and agent auto-registration events has been decreased from 365 days to 1 day. When upgrading from previous versions, it is recommended to decrease these values to 1 day manually as it can improve Zabbix server performance.

## Possible issues with SELinux

Socket-based inter-process communication has been added in Zabbix 3.4. On systems where SELinux is enabled it may be required to add SELinux rules to allow Zabbix create/use UNIX domain sockets in the `SocketDir` directory. Currently socket files are used by server (alerter, preprocessing, IPMI) and proxy (IPMI). Socket files are persistent, meaning are present while the process is running.

## See also

- [Template changes](#)

From:

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

Permanent link:

[https://www.zabbix.com/documentation/3.4/manual/installation/upgrade\\_notes\\_340?rev=1549953400](https://www.zabbix.com/documentation/3.4/manual/installation/upgrade_notes_340?rev=1549953400)

Last update: **2019/02/12 06:36**

