

ACC: Support preprocessing for LLD rules

- [Zabbix Acceptance](#)
- [Blog post](#)
- [Use cases](#)
- [Related tasks](#)
- [Out of scope](#)
- [Changelog](#)



The document is ready to be reviewed by spec team, partners and community members.

Feature requests: [ZBXNEXT-4087](#), [ZBXNEXT-4877](#)

Zabbix Acceptance

1. LLD rules must support preprocessing functionality like it is currently supported for items and item prototypes
 1. Only the following preprocessing types will be supported
 1. Text > Regular expression
 2. Structured data > JSON Path
 3. Validation > Check for error in JSON
 4. Validation > Check for error using regular expression
 5. Throttling > Discard unchanged
 6. Throttling > Discard unchanged with heartbeat
 2. It will also support preprocessing actions relevant for supported preprocessing types
2. ([ZBXNEXT-4877](#)) LLD will accept normal JSON containing an array, i.e. without 'data' object
 1. Zabbix will automatically extract macro and value if an array field uses `{#MACRO}` syntax as a key
 2. For backward compatibility Zabbix will accept JSON notation with "data" element.
 1. If JSON contains object with only one "data" array element, then it will automatically extract content of the element using JSON Path `$.data`
3. ([ZBXNEXT-4877](#)) LLD will accept optional user-defined list of LLD macros
 1. LLD rule configuration form will be extended to support a new tab to configure LLD macros
 2. A set of `{#MACRO}=<json path>` can be defined, for example: `{#NAME}=$.name`, `{#FS}=$.fs.name`, `{#MOUNT}=$.fs.mountpoint`
 1. JSON keys with any special characters and unicode must be supported using optional square bracket notation, like `$['unicode + special chars #1']`['unicode + special chars #2']
 3. User defined LLD macros will override macros found in JSON (in reality it should not happen)
 4. Supported syntax for JSON path must be documented
4. LLD filters must be applied after preprocessing rules
5. ([ZBXNEXT-4877](#)) The "data" element will be removed from all items related to discovery (agent, jmx, snmp, odbc, etc)
 1. Any new native discovery checks will use new syntax without the "data" elements

Blog post

A blog post on blog.zabbix.com must be created to explain why this functionality is needed, how it works and describe how to use it for various use cases.

Use cases

1. Use normal JSON as a native data source for LLD, therefore it will handle JSON without any preprocessing

1. Discovery from array of JSON objects with nested objects
2. I do not want Zabbix to execute LLD rules if it receives identical values

Related tasks

1. ACC: Preprocessing rule for UTF8 encoding

Out of scope

1. Support of mass update for preprocessing rules
2. Merging of preprocessing rules and filters if it will ever be considered

Changelog

1.0rc4

1. Support of CSV to JSON moved to a separate document
2. Only 5 preprocessing types will be supported for LLD rules
3. LLD rules will support new syntax of JSON without "data" element, while continue to process old one
4. Removed autocomplete functionality. Reasoning: it is not a trivial thing and related also to other parts of UI, let's keep it separate.

1.0rc5

1. Removed use case with CSV as an input
2. Added missing Throttling Discard unchanged with heartbeat
3. Removed section Open questions

1.1

1. Added requirement for a blog post

1.2

1. JSON path in LLD extracted from [ZBXNEXT-4087](#) in to [ZBXNEXT-4877](#)