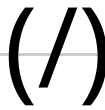




---

[Docs \(/guide\)](#)

---

[Docs \(/guide\)](#)

You are looking at documentation for an older release. Not what you want? See the current release documentation ([../current/index.html](#)).

[Logstash Reference \[5.3\] \(index.html\)](#) » [Filter plugins \(filter-plugins.html\)](#) » [mutate](#)

« [metrics \(plugins-filters-metrics.html\)](#)

[oui](#) » [\(plugins-filters-oui.html\)](#)

## mutate

- Version: 3.1.3
- Released on: September 29, 2016
- Changelog (<https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb>)

## Getting Help

For questions about the plugin, open a topic in the Discuss (<http://discuss.elastic.co>) forums. For bugs or feature requests, open an issue in Github (<https://github.com/elastic/logstash>). For the list of Elastic supported plugins, please consult the Elastic Support Matrix ([https://www.elastic.co/support/matrix#show\\_logstash\\_plugins](https://www.elastic.co/support/matrix#show_logstash_plugins)).

## Description

The mutate filter allows you to perform general mutations on fields. You can rename, remove, replace, and modify fields in your events.

## Synopsis

This plugin supports the following configuration options:

Required configuration options:

```
mutate {
}
```

Available configuration options:

Setting	Input type	Required
<code>add_field</code> (plugins-filters-mutate.html#plugins-filters-mutate-add_field)	hash (configuration-file-structure.html#hash)	No
<code>add_tag</code> (plugins-filters-mutate.html#plugins-filters-mutate-add_tag)	array (configuration-file-structure.html#array)	No
<code>convert</code> (plugins-filters-mutate.html#plugins-filters-mutate-convert)	hash (configuration-file-structure.html#hash)	No
<code>enable_metric</code> (plugins-filters-mutate.html#plugins-filters-mutate-enable_metric)	boolean (configuration-file-structure.html#boolean)	No
<code>gsub</code> (plugins-filters-mutate.html#plugins-filters-mutate-gsub)	array (configuration-file-structure.html#array)	No
<code>id</code> (plugins-filters-mutate.html#plugins-filters-mutate-id)	string (configuration-file-structure.html#string)	No
<code>join</code> (plugins-filters-mutate.html#plugins-filters-mutate-join)	hash (configuration-file-structure.html#hash)	No
<code>lowercase</code> (plugins-filters-mutate.html#plugins-filters-mutate-lowercase)	array (configuration-file-structure.html#array)	No
<code>merge</code> (plugins-filters-mutate.html#plugins-filters-mutate-merge)	hash (configuration-file-structure.html#hash)	No
<code>periodic_flush</code> (plugins-filters-mutate.html#plugins-filters-mutate-periodic_flush)	boolean (configuration-file-structure.html#boolean)	No

Setting	Input type	Required
<code>remove_field</code> (plugins-filters-mutate.html#plugins-filters-mutate-remove_field)	array (configuration-file-structure.html#array)	No
<code>remove_tag</code> (plugins-filters-mutate.html#plugins-filters-mutate-remove_tag)	array (configuration-file-structure.html#array)	No
<code>rename</code> (plugins-filters-mutate.html#plugins-filters-mutate-rename)	hash (configuration-file-structure.html#hash)	No
<code>replace</code> (plugins-filters-mutate.html#plugins-filters-mutate-replace)	hash (configuration-file-structure.html#hash)	No
<code>split</code> (plugins-filters-mutate.html#plugins-filters-mutate-split)	hash (configuration-file-structure.html#hash)	No
<code>strip</code> (plugins-filters-mutate.html#plugins-filters-mutate-strip)	array (configuration-file-structure.html#array)	No
<code>update</code> (plugins-filters-mutate.html#plugins-filters-mutate-update)	hash (configuration-file-structure.html#hash)	No
<code>uppercase</code> (plugins-filters-mutate.html#plugins-filters-mutate-uppercase)	array (configuration-file-structure.html#array)	No

## Details

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

### add\_field

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash (configuration-file-structure.html#hash)
- Default value is `{}`

If this filter is successful, add any arbitrary fields to this event. Field names can be dynamic and include parts of the event using the `%{field}`.

Example:

```
filter {
  mutate {
    add_field => { "foo_%{somefield}" => "Hello world, from %{host}" }
  }
}
```

```
# You can also add multiple fields at once:
filter {
  mutate {
    add_field => {
      "foo_%{somefield}" => "Hello world, from %{host}"
      "new_field" => "new_static_value"
    }
  }
}
```

If the event has field "somefield" == "hello" this filter, on success, would add field foo\_hello if it is present, with the value above and the %{host} piece replaced with that value from the event. The second example would also add a hardcoded field.

## add\_tag

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array (configuration-file-structure.html#array)
- Default value is []

If this filter is successful, add arbitrary tags to the event. Tags can be dynamic and include parts of the event using the %{field} syntax.

Example:

```
filter {
  mutate {
    add_tag => [ "foo_%{somefield}" ]
  }
}
```

```
# You can also add multiple tags at once:
filter {
  mutate {
    add_tag => [ "foo_%{somefield}", "taggedy_tag" ]
  }
}
```

If the event has field "somefield" == "hello" this filter, on success, would add a tag foo\_hello (and the second example would of course add a taggedy\_tag tag).

## convert

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash (configuration-file-structure.html#hash)
- There is no default value for this setting.

Convert a field's value to a different type, like turning a string to an integer. If the field value is an array, all members will be converted. If the field is a hash, no action will be taken.

If the conversion type is `boolean`, the acceptable values are:

- **True:** `true`, `t`, `yes`, `y`, and `1`
- **False:** `false`, `f`, `no`, `n`, and `0`

If a value other than these is provided, it will pass straight through and log a warning message.

Valid conversion targets are: integer, float, string, and boolean.

Example:

```
filter {  
  mutate {  
    convert => { "fieldname" => "integer" }  
  }  
}
```

## enable\_metric

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is boolean (configuration-file-structure.html#boolean)
- Default value is `true`

Disable or enable metric logging for this specific plugin instance by default we record all the metrics we can, but you can disable metrics collection for a specific plugin.

## gsub

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array (configuration-file-structure.html#array)
- There is no default value for this setting.

Convert a string field by applying a regular expression and a replacement. If the field is not a string, no action will be taken.

This configuration takes an array consisting of 3 elements per field/substitution.

Be aware of escaping any backslash in the config file.

Example:

```
filter {
  mutate {
    gsub => [
      # replace all forward slashes with underscore
      "fieldname", "/", "_",
      # replace backslashes, question marks, hashes, and minuses
      # with a dot "."
      "fieldname2", "[\\?#-]", "."
    ]
  }
}
```

## id

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is string ([configuration-file-structure.html#string](#))
- There is no default value for this setting.

Add a unique ID to the plugin configuration. If no ID is specified, Logstash will generate one. It is strongly recommended to set this ID in your configuration. This is particularly useful when you have two or more plugins of the same type, for example, if you have 2 grok filters. Adding a named ID in this case will help in monitoring Logstash when using the monitoring APIs.

```
output {
  stdout {
    id => "my_plugin_id"
  }
}
```

## join

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash ([configuration-file-structure.html#hash](#))
- There is no default value for this setting.

Join an array with a separator character. Does nothing on non-array fields.

Example:

```
filter {
  mutate {
    join => { "fieldname" => "," }
  }
}
```

## lowercase

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array ([configuration-file-structure.html#array](#))

- There is no default value for this setting.

Convert a string to its lowercase equivalent.

Example:

```
filter {  
  mutate {  
    lowercase => [ "fieldname" ]  
  }  
}
```

## merge

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash (<configuration-file-structure.html#hash>)
- There is no default value for this setting.

Merge two fields of arrays or hashes. String fields will be automatically be converted into an array, so:

```
`array` + `string` will work  
`string` + `string` will result in an 2 entry array in `dest_field`  
`array` and `hash` will not work
```

Example:

```
filter {  
  mutate {  
    merge => { "dest_field" => "added_field" }  
  }  
}
```

## periodic\_flush

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is boolean (<configuration-file-structure.html#boolean>)
- Default value is `false`

Call the filter flush method at regular interval. Optional.

## remove\_field

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array (<configuration-file-structure.html#array>)
- Default value is `[]`

If this filter is successful, remove arbitrary fields from this event. Example:

```
filter {
  mutate {
    remove_field => [ "foo_%{somefield}" ]
  }
}
```

```
# You can also remove multiple fields at once:
filter {
  mutate {
    remove_field => [ "foo_%{somefield}", "my_extraneous_field" ]
  }
}
```

If the event has field "somefield" == "hello" this filter, on success, would remove the field with name `foo_hello` if it is present. The second example would remove an additional, non-dynamic field.

## remove\_tag

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array ([configuration-file-structure.html#array](#))
- Default value is []

If this filter is successful, remove arbitrary tags from the event. Tags can be dynamic and include parts of the event using the `%{field}` syntax.

Example:

```
filter {
  mutate {
    remove_tag => [ "foo_%{somefield}" ]
  }
}
```

```
# You can also remove multiple tags at once:
filter {
  mutate {
    remove_tag => [ "foo_%{somefield}", "sad_unwanted_tag" ]
  }
}
```

If the event has field "somefield" == "hello" this filter, on success, would remove the tag `foo_hello` if it is present. The second example would remove a sad, unwanted tag as well.

## rename

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash ([configuration-file-structure.html#hash](#))
- There is no default value for this setting.



Rename one or more fields.

Example:

```
filter {  
  mutate {  
    # Renames the 'HOSTORIP' field to 'client_ip'  
    rename => { "HOSTORIP" => "client_ip" }  
  }  
}
```

## replace

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash (configuration-file-structure.html#hash)
- There is no default value for this setting.

Replace a field with a new value. The new value can include `%{foo}` strings to help you build a new value from other parts of the event.

Example:

```
filter {  
  mutate {  
    replace => { "message" => "%{source_host}: My new message" }  
  }  
}
```

## split

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash (configuration-file-structure.html#hash)
- There is no default value for this setting.

Split a field to an array using a separator character. Only works on string fields.

Example:

```
filter {  
  mutate {  
    split => { "fieldname" => "," }  
  }  
}
```

## strip

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array (configuration-file-structure.html#array)
- There is no default value for this setting.

Strip whitespace from field. NOTE: this only works on leading and trailing whitespace.

Example:

```
filter {  
  mutate {  
    strip => ["field1", "field2"]  
  }  
}
```

## update

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is hash ([configuration-file-structure.html#hash](#))
- There is no default value for this setting.

Update an existing field with a new value. If the field does not exist, then no action will be taken.

Example:

```
filter {  
  mutate {  
    update => { "sample" => "My new message" }  
  }  
}
```

## uppercase

[edit \(https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb\)](https://github.com/logstash-plugins/logstash-filter-mutate/edit/master/lib/logstash/filters/mutate.rb)

- Value type is array ([configuration-file-structure.html#array](#))
- There is no default value for this setting.

Convert a string to its uppercase equivalent.

Example:

```
filter {  
  mutate {  
    uppercase => [ "fieldname" ]  
  }  
}
```

---

[« metrics \(plugins-filters-metrics.html\)](#)

[oui » \(plugins-filters-oui.html\)](#)

# Top Videos

- Elasticsearch Demo (<https://www.elastic.co/webinars/getting-started-elasticsearch?baymax=default&elektra=docs&storm=top-video>)
- Kibana 101 (<https://www.elastic.co/webinars/getting-started-kibana?baymax=default&elektra=docs&storm=top-video>)
- Logstash Primer (<https://www.elastic.co/webinars/getting-started-logstash?baymax=default&elektra=docs&storm=top-video>)

## On this page

Getting Help

Description

Synopsis

Details

Logstash Reference: 5.3	▼
Logstash Introduction (introduction.html)	
Getting Started with Logstash (getting-started-with-logstash.html)	
How Logstash Works (pipeline.html)	
Setting Up and Running Logstash (setup-logstash.html)	
Breaking changes (breaking-changes.html)	
Upgrading Logstash (upgrading-logstash.html)	
Configuring Logstash (configuration.html)	
Working with Filebeat Modules (filebeat-modules.html)	
Deploying and Scaling Logstash (deploying-and-scaling.html)	
Performance Tuning (performance-tuning.html)	
Monitoring APIs (monitoring.html)	
Working with plugins (working-with-plugins.html)	
Input plugins (input-plugins.html)	
Output plugins (output-plugins.html)	
Filter plugins (filter-plugins.html)	
age (plugins-filters-age.html)	
aggregate (plugins-filters-aggregate.html)	
alter (plugins-filters-alter.html)	