

Graylog: Getting started

How to start using Graylog

Welcome on Stackhero's documentation!

Stackhero provides Graylog instances that are ready for production in just 2 minutes!

Including TLS encryption (aka HTTPS), customizable domain name, email server, backups and updates in just a click.

Try our [managed Graylog cloud](#) in just 2 minutes

- [Create a first input in Graylog](#)
- [Graylog codes examples](#)
- [Send Rsyslog logs to Graylog using TLS encryption](#)
- [Error "failed to parse field \[XXXX\] of type \[YYYY\]"](#)
 - [Update the Elasticsearch mapping](#)
- [Error "Unable to write audit log entry"](#)

Create a first input in Graylog

An "input" is where Graylog will get your logs.

You can send your logs to Graylog using TCP or UDP.

Graylog can also obtain your logs from an API, a Kafka queue, a RabbitMQ server and a lot of other methods.

For this example, We will create a raw UDP input.

On Graylog interface, go to "System" then "Inputs".

Select "Raw/Plaintext UDP" and click on "Launch new input".

Configure your input like this and valid the form:

- Node: select your node
- Title: RAW UDP
- Port: 5555

On your computer, open a terminal and send an UDP message to your Graylog server (don't forget to replace XXXXXX per your service domain name):

From macOS: `echo "Hello Graylog from UDP" | nc -u -w1 -c XXXXXX.stackhero-network.com 5555`

From Linux: `echo "Hello Graylog from UDP" | nc -u -w1 XXXXXX.stackhero-network.com 5555`

Go back to Graylog and click on "Search": you should see your message 

Congrats, you have sent your first message to Graylog!

Now you can create some real inputs and dashboards.

To help you, we recommend to use the [Graylog's official documentation](#).

Graylog codes examples

You will find some codes examples in our [git repository](#)

Send Rsyslog logs to Graylog using TLS encryption

You have a Rsyslog client and want to send your logs to Graylog, in a secured way.

Here is how to it:

- First, create a "Syslog TCP" input on Graylog. Give it a title and validate the form.
⚠️ Do not activate any TLS option on Graylog's input. TLS will be handled directly by a reverse proxy on your instance and will not be handled by Graylog.
- Then, go to your Graylog service configuration in Stackhero dashboard and enable the "TLS encryption" for the Syslog TCP port 514.
- Finally, configure your Rsyslog as follow, and replace `XXXXXX.stackhero-network.com` per your instance hostname:

```
# Define TLS CA certificate
global(
  DefaultNetstreamDriver="gtls"
  DefaultNetstreamDriverCAFile="/etc/ssl/certs/ca-certificates.crt"
)

# Send all logs to a remote server.
# An on-disk queue is created for this action. If the remote host is
# down, messages are spooled to disk and sent when it is up again.
# See https://www.rsyslog.com/doc/v8-stable/configuration/actions.html
# And https://www.rsyslog.com/doc/v8-stable/configuration/modules/omfwd.html
*. * action(
  type="omfwd"
  target="XXXXXX.stackhero-network.com"
  port="514"
  protocol="tcp"
  KeepAlive="on"
  KeepAlive.Interval="30"
  StreamDriver="gtls"
  StreamDriverMode="1"
  StreamDriverAuthMode="x509/name"
  ResendLastMSGOnReconnect="on"
  queue.filename="fwdRule1" # unique name prefix for spool files
  queue.type="LinkedList"
  queue.maxDiskSpace="256m"
  queue.saveOnShutdown="on"
  action.resumeRetryCount="-1"
  action.resumeInterval="30"
```

- Restart your Rsyslog server and try to send a log with the command `logger This is a test`.

That's it, you are now sending your logs directly to Graylog in a secure way, using TLS encryption!

Error "failed to parse field [XXXX] of type [YYYY]"

Maybe you will get an error like "org.elasticsearch.index.mapper.MapperParsingException: failed to parse field [time] of type [long] in document with id 'xxxx'".

You can see this error in "logs", available in Stackhero dashboard, or in your Graylog admin panel, in "System" / "Overview" / "Indexer failures".

This error means that you sent a log with a value for the field `time` that was a different type that Elasticsearch was waiting for (in that case, a "long" type).

Graylog uses the dynamic mapping feature of Elasticsearch. When you first send a log, Elasticsearch will try to guess the type of the fields.

So if you send a log containing a field `time` with `1234` as a value, Elasticsearch will define this field type as a numeric one (long).

If you send another log then with the field `time` set as `abcd`, which is a string this time, Elasticsearch will reject it because it expected to have a numeric value.

Note that the field name `time` is just an example, as the `long` type. It could be any field name and any type.

To resolve this issue, you have to redefine the type that Elasticsearch is waiting for.

You will get more informations on this on the official [Graylog documentation](#).

To update the Elasticsearch mapping, simply go to the new article.

Update the Elasticsearch mapping

You have first to activate the Elasticsearch access in Stackhero dashboard.

Go to your Graylog service, then "configure", and activate the Elasticsearch access.

⚠ Be careful with what you are doing here as you can totally block your Elasticsearch and even lost your data!
If you don't understand what you are doing, don't do it.

First, we will define our new mapping. Here, we redefine the field `time` to the type `string`.

You will get a list of available types in [Elasticsearch field datatypes documentation](#).

Edit as wanted and save this content to a file named `graylog-custom-mapping.json`:

```
{
```

```
"template": "graylog_*",
"mappings" : {
  "message" : {
    "properties" : {
      "time" : {
        "type" : "string",
        "index" : "not_analyzed"
      }
    }
  }
}
```

Then, post this file with this curl command: `curl -u 'admin' -X PUT -d '@graylog-custom-mapping.json' -H 'Content-Type: application/json' 'https://XXXXXX.stackhero-network.com/elasticsearch/_template/graylog-custom-mapping?pretty'`

Don't forget to replace `XXXXXX.stackhero-network.com` per your instance domain name.

You should have this reply:

```
{
  "acknowledged" : true
}
```

Finally, you can check that the mapping has been updated with this command: `curl -u 'admin' -X GET 'https://XXXXXX.stackhero-network.com/elasticsearch/graylog_deflector/_mapping?pretty'`

Error "Unable to write audit log entry"

If you see an error like `Unable to write audit log entry because there is no valid license` or `Not running cleanup for auditlog entries in MongoDB because there is no valid license` it is because you have activated Graylog Enterprise without entering a license.

If you have a license, enter it in your Graylog interface.

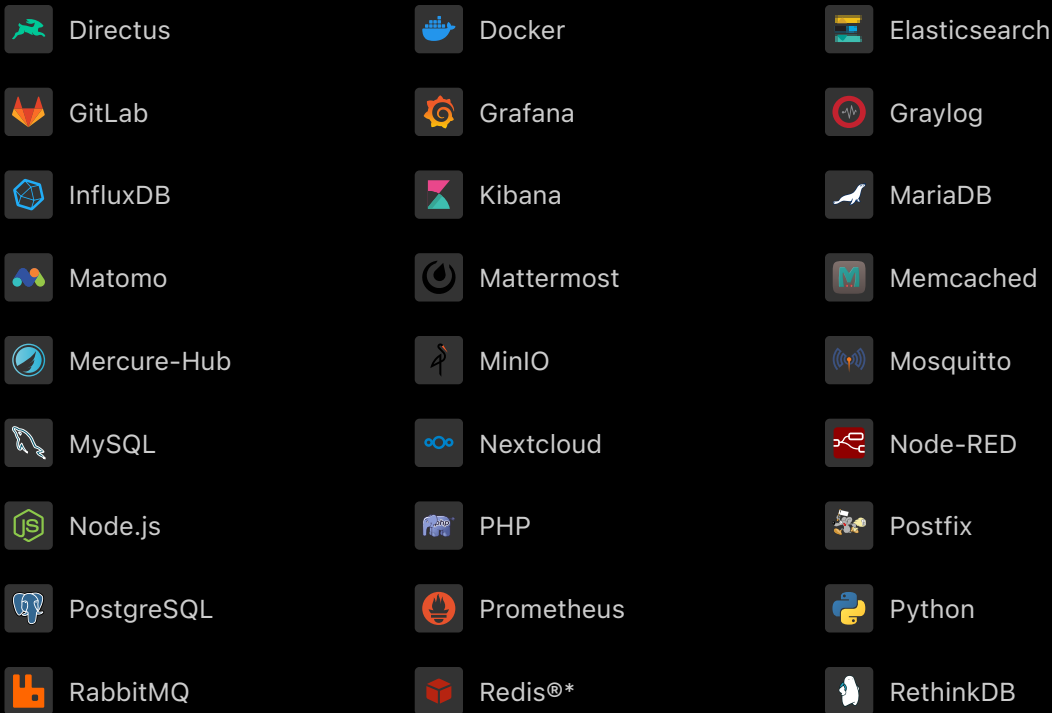
If you don't have a license, simply disable Graylog Enterprise in Stackhero dashboard.

Other articles about Graylog that might interest you


- [Choose inputs types](#)
How to choose the right Graylog input type
- [Configure inputs](#)
How to configure Graylog inputs
- [Handle retention](#)
How to configure retention

- [Alerting](#)
How to send Graylog alerts by email, Slack or Mattermost
- [Enterprise license](#)
How to handle Graylog Enterprise license
- [Using with Node.js](#)
How to send logs from Node.js to Graylog
- [Using with Dot NET](#)
How to send logs from .NET/Serilog to Graylog
- [Using with Python](#)
How to send logs from Python to Graylog

Our Managed Services



- [Terms of Service](#)
- [Privacy Policy](#)
- [Documentations](#)
- [Support](#)
- [Status](#)

 [English](#)

 [Global](#)



Directus, Docker, Elasticsearch, GitLab, Grafana, Graylog, InfluxDB, Kibana, MariaDB, Matomo, Mattermost, Memcached, Mercure-Hub, MinIO, MongoDB, Mosquitto, MySQL, Nextcloud, Node-RED, Node.js, PHP, Postfix, PostgreSQL, Prometheus, Python, RabbitMQ, Redis*, RethinkDB are trademarks and property of their respective owners. All product and service names used on this website are for identification purposes of their open sourced products only and do not imply endorsement. Stackhero is not affiliated to these trademarks or companies.

*Redis is a registered trademark of Redis Ltd. Any rights therein are reserved to Redis Ltd. Any use by Stackhero is for referential purposes only and does not indicate any sponsorship, endorsement or affiliation between Redis and Stackhero

Some icons of this website are made by Dimitry Miroliubov.

© Stackhero. All rights reserved.