

Vector

Coralogix provides seamless integration with Vector so you can send your logs from anywhere and parse them according to your needs.

Prerequisites

- Vector [installed](#)
- For those customers with a [Kubernetes](#) environment, Helm 2.9+ Package Manager [installed](#)

Usage

You must input the following variables when creating a Coralogix logger instance or updating your configuration file:

- ``api_key`` : Input your Coralogix [Send-Your-Data API key](#).
- `applicationName` and `subsystemName` : Input [application and subsystem names](#).
- `coralogix_domain` : Input the Coralogix [domain](#) associated with your account.

Configuration for Non-Kubernetes Environments

STEP 1. Edit default configuration file

Edit the default configuration file `vector.toml` and paste the following configuration below. Update the following:

- `event.log.applicationName` = "VECTOR_APP"
- `event.log.subsystemName` = "VECTOR_SUBSYS"
- `uri` = [REST API Singles endpoint](#) associated with your Coralogix [domain](#)
- `headers.authorization` = "Bearer <<API_KEY>> "

```
# This input is just for test or demo purposes and is supported only since  
Vector v0.12.0. Feel free to use any other input.
```

```
[sources.my_source_id]  
  type = "demo_logs" # required  
  lines = [{"\msg\":"success\"}, {"\code\":"200"}], [{"\msg\":"error\"},  
\code\":"500"}] # optional, no default, relevant when [format](#format) =
```

```

`shuffle`
    sequence = false # optional, default, relevant when [`format`](#format) =
`shuffle`
    batch_interval = 10
    format = "shuffle"

[transforms.prepare_for_coralogix]
    # General
    type = "lua" # required
    inputs = ["my_source_id"] # required
    version = "2" # required
    # Hooks
    hooks.process = '''
function (event, emit)
    event.log = {
        applicationName = "VECTOR_APP",
        subsystemName = "VECTOR_SUBSYS",
        json = event.log,
        computerName = os.getenv("HOSTNAME"),
        timestamp = os.time() * 1000
    }
    emit(event)
end
'''

[sinks.coralogix]
    # General
    type = "http" # required
    inputs = ["prepare_for_coralogix"] # required
    compression = "none" # optional, default
    method = "post"
    uri = https://ingress.eu2.coralogix.com/logs/v1/singles # required
    # Batch
    batch.max_bytes = 1049000 # optional, default, bytes
    batch.timeout_secs = 2 # optional, default, seconds
    # Encoding
    encoding.codec = "json" # required
    # Healthcheck
    #healthcheck.enabled = true # optional, default
    headers.Content-Type = "application/json"
    headers.authorization = "Bearer <<API_key>>"

```

STEP 2. Add your own sources

That code will allow you to generate the test traffic. After testing the connection, add your own sources, as in the example below.

```
[sources.my_source_id]
  type = "file"
  include = ["/var/log/access.log "]
```

You can read more about this configuration [here](#).

STEP 3. Restart Vector

Remember to restart Vector after every change to load updated configuration.

Configuration for Kubernetes Environments

STEP 1. Add Helm chart repo

```
helm repo add vector https://helm.vector.dev
helm repo update
```

STEP 2. Create an override.yaml file

Prior to installation create an override.yaml file and paste the following configuration below. Remember to update:

- event.log.applicationName: "VECTOR_APP"
- event.log.subsystemName: "VECTOR_SUBSYS"
- uri: [logs endpoint](#) associated with your Coralogix domain
- headers.private_key: <<API_KEY>>

```
# vector override.yaml
---
role: "Agent"

customConfig:
  sources:
    kubernetes_logs:
      type: kubernetes_logs

  transforms:
```

```
prepare_for_coralogix:
  type: lua
  inputs: ["kubernetes_logs"]
  version: "2"
  #hooks
  hooks:
    process: |
      function (event, emit)
        event.log = {
          applicationName = "VECTOR_APP",
          subsystemName = "VECTOR_SUBSYS",
          json = event.log,
          computerName = os.getenv("HOSTNAME"),
          timestamp = os.time() * 1000
        }
        emit(event)
      end

sinks:
  coralogix:
    type: http
    inputs: ["prepare_for_coralogix"]
    compression: none
    uri: https://ingress.eu2.coralogix.com/logs/v1/singles
    batch:
      maxbytes:
        1049000
      timeout_secs:
        2
    encoding:
      codec:
        json
    request:
      headers:
        Content-Type: application/json
        private_key: <<api_key>>
```

STEP 3. Deploy the chart

```
helm install vector-agent vector/vector -f override.yaml
```



STEP 4. Remove the Daemonset

```
helm uninstall vector-agent
```



Support

Need help?

Our world-class customer success team is available 24/7 to walk you through your setup and answer any questions that may come up.

Feel free to reach out to us **via our in-app chat** or by sending us an email at support@coralogix.com.

 Last updated: July 1, 2025

Was this helpful?

Leave your feedback here.

☐ **Yes** ☐ **No**

Send

© 2025 Coralogix. All rights reserved.

Generated on: November 17, 2025

Source: <https://coralogix.com/docs/integrations/files/vector/>