



Troubleshooting the OpenTelemetry Collector with Elastic

Andrzej Stencel

Elastic Berlin Meetup, March 2025

Hi, my name is:

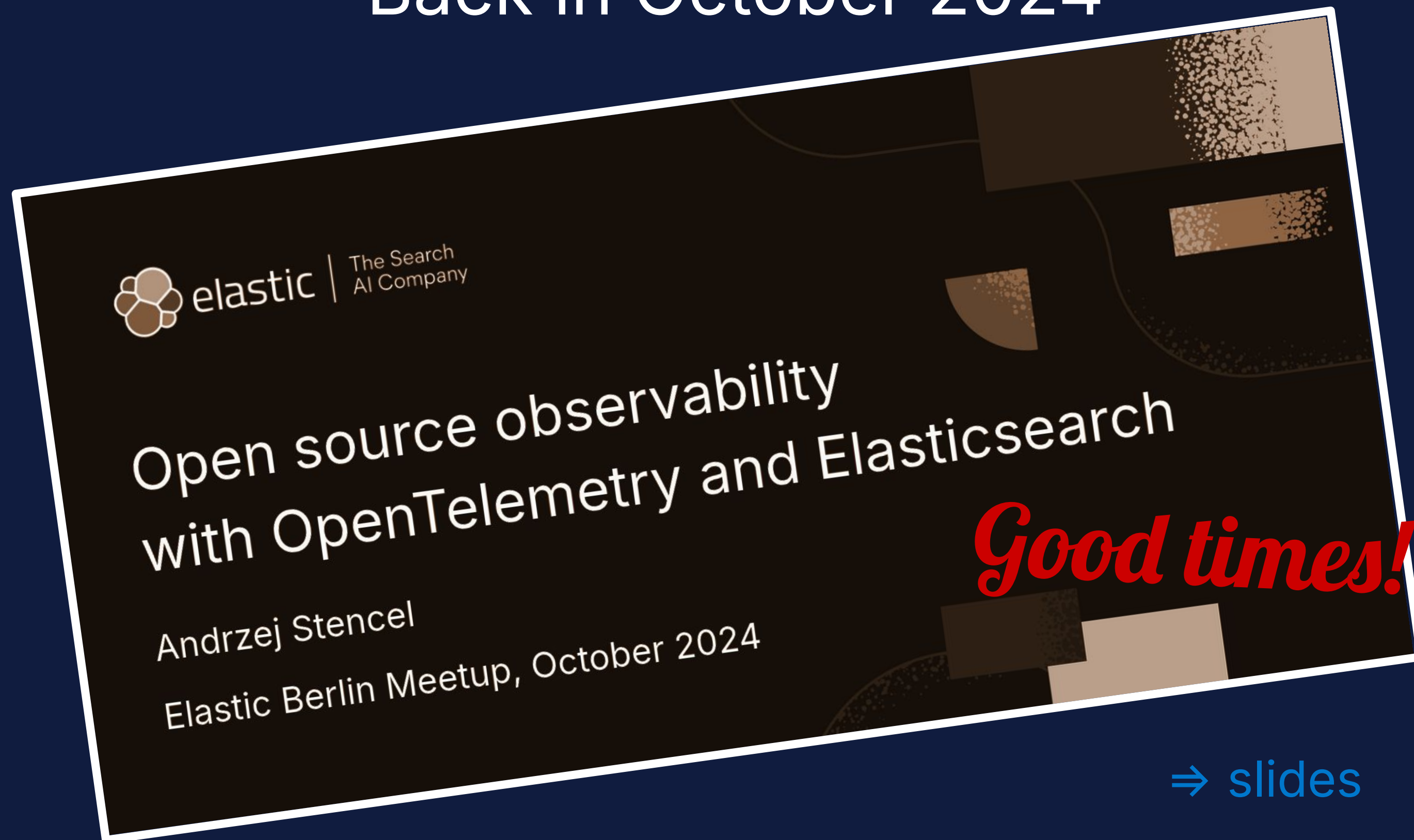
Andrzej Stencel

Senior Software Engineer at Elastic

Maintainer of [OpenTelemetry Collector Contrib](#)



Back in October 2024



⇒ slides

EDOT Collector = Elastic Distribution of the OpenTelemetry Collector

```
$ curl -LO  
https://artifacts.elastic.co/downloads/beats/elastic-  
-agent/elastic-agent-8.17.4-linux-x86_64.tar.gz  
  
$ tar zxvf elastic-agent-8.17.4-linux-x86_64.tar.gz  
  
$ cd elastic-agent-8.17.4-linux-x86_64  
  
$ sudo ./elastic-agent otel
```


Troubleshooting the EDOT Collector (or any OpenTelemetry Collector distro)

Troubleshooting:

Finding out if something is doing what I think it should be doing and when it's not doing it, why - so that I can fix it and make it do what I want.

Example configuration file

```
service:
  pipelines:
    logs:
      receivers:
        - filelog/pings
      exporters:
        - elasticsearch
receivers:
  filelog/pings:
    include:
      - pings.log
    start_at: end
exporters:
  elasticsearch:
    endpoint: ${env:ELASTIC_ENDPOINT}
    api_key: ${env:ELASTIC_API_KEY}
    mapping:
      mode: otel
    flush:
      interval: 1s
```

\$ ping -D 8.8.8.8 | tee pings.log

PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.

[1742934326.645847] 64 bytes from 8.8.8.8: icmp_seq=1 ttl=255 time=25.0 ms

[1742934327.669739] 64 bytes from 8.8.8.8: icmp_seq=2 ttl=255 time=32.5 ms

[1742934328.717848] 64 bytes from 8.8.8.8: icmp_seq=3 ttl=255 time=31.5 ms

[1742934329.730502] 64 bytes from 8.8.8.8: icmp_seq=4 ttl=255 time=30.7 ms

Troubleshooting with logs

```
$ sudo ./elastic-agent otel --config ../01-logs-from-files-to-elasticsearch.yaml
Starting in otel mode
2025-03-25T21:57:59.525+0100 info service@v0.119.0/service.go:186 Setting up own
telemetry...
2025-03-25T21:57:59.525+0100 warn elasticsearchexporter@v0.119.0/config.go:367 dedot has been
deprecated: in the future, dedotting will always be performed in ECS mode only{"kind":
"exporter", "data_type": "logs", "name": "elasticsearch"}
2025-03-25T21:57:59.526+0100 info service@v0.119.0/service.go:252 Starting
./elastic-agent... {"Version": "8.17.4", "NumCPU": 20}
2025-03-25T21:57:59.526+0100 info extensions/extensions.go:39 Starting extensions...
2025-03-25T21:57:59.526+0100 info adapter/receiver.go:41 Starting stanza receiver
{"kind": "receiver", "name": "filelog/pings", "data_type": "logs"}
2025-03-25T21:57:59.526+0100 warn fileconsumer/file.go:49 finding files {"kind":
"receiver", "name": "filelog/pings", "data_type": "logs", "component": "fileconsumer",
"error": "no files match the configured criteria"}
2025-03-25T21:57:59.526+0100 info service@v0.119.0/service.go:275 Everything is ready. Begin
running and processing data.
```

Customizing collector's logs

```
service:
  telemetry:
    logs:
      level: debug # default: info. other values: warn, error
      sampling:
        enabled: false # default: true
      disable_caller: true # default: false
      encoding: json # default: console
      output_paths: # default: ["stderr"]
        - stdout
        - ./collector-output.log
      error_output_paths: # default: ["stderr"]
        - stderr
        - ./collector-errors.log
```

<https://opentelemetry.io/docs/collector/internal-telemetry/#configure-internal-logs>

Collector's logs

Useful for discovering:

- configuration errors, connection errors, data loss errors
- deprecation warnings
- basic information about what the collector's components are doing
 - like the File Log receiver starting to watch a file

Troubleshooting with metrics

```
$ curl -s http://localhost:8888/metrics | grep '^otelcol_[er]'
```

otelcol_exporter_send_failed_log_records{exporter="elasticsearch",service_instance_id="491f5257-2538-494c-9be5-41b712c108c3",service_name="./elastic-agent-8.17.4-linux-x86_64/elastic-agent",service_version="8.17.4"} 0

otelcol_exporter_sent_log_records{exporter="elasticsearch",service_instance_id="491f5257-2538-494c-9be5-41b712c108c3",service_name="./elastic-agent-8.17.4-linux-x86_64/elastic-agent",service_version="8.17.4"} 104

otelcol_receiver_accepted_log_records{receiver="filelog/pings",service_instance_id="491f5257-2538-494c-9be5-41b712c108c3",service_name="./elastic-agent-8.17.4-linux-x86_64/elastic-agent",service_version="8.17.4",transport=""} 104

otelcol_receiver_refused_log_records{receiver="filelog/pings",service_instance_id="491f5257-2538-494c-9be5-41b712c108c3",service_name="./elastic-agent-8.17.4-linux-x86_64/elastic-agent",service_version="8.17.4",transport=""} 0

Customizing collector's metrics

```
service:  
  telemetry:  
    metrics:  
      level: detailed # default: normal, other values: none, basic  
      address: 0.0.0.0:8889 # default: localhost:8888
```

<https://opentelemetry.io/docs/collector/internal-telemetry/#configure-internal-metrics>

✨ New: Send collector's metrics via OTLP ✨

Expose in Prometheus format:

```
service:
  telemetry:
    metrics:
      readers:
        - pull:
            exporter:
              prometheus:
                host: '0.0.0.0'
                port: 8889
```

Send to an OTLP endpoint:

```
service:
  telemetry:
    metrics:
      readers:
        - periodic:
            exporter:
              otlp:
                protocol: http/protobuf
                endpoint: https://backend:4318
```

<https://opentelemetry.io/docs/collector/internal-telemetry/#configure-internal-metrics>

✨ New: Send collector's logs via OTLP ✨

```
service:
  telemetry:
    logs:
      processors:
        - batch:
            exporter:
              otlp:
                protocol: http/protobuf
                endpoint: https://backend:4318
```

<https://opentelemetry.io/docs/collector/internal-telemetry/#configure-internal-logs>

✨ New: Send collector's traces via OTLP ✨

```
service:
  telemetry:
    traces:
      processors:
        - batch:
            exporter:
              otlp:
                protocol: http/protobuf
                endpoint: https://backend:4318
```

<https://opentelemetry.io/docs/collector/internal-telemetry/#configure-internal-traces>

Monitoring the OpenTelemetry Collector via OTLP

- Send the collector's logs, metrics, traces to an OTLP endpoint
 - Ideally a separate process from the one being observed
- Collect those signals and export to Elasticsearch
- Analyze those signals and alert on them
- ⇒ ⇒ ⇒ Work in progress ⇒ ⇒ ⇒

Elastic + OpenTelemetry = ❤️

Elastic is among the top contributors to OpenTelemetry

<https://elastic.github.io/opentelemetry/>

<https://www.elastic.co/observability-labs/blog>

Thank you!

Slides:

<https://andrzej-stencel.github.io/2025/03/26/elastic-berlin-meetup.html>

