

<https://github.com/signalfx/splunk-otel-collector/releases>

```
<dependency>
  <groupId>io.micrometer</groupId>
  <artifactId>micrometer-tracing-bridge-otel</artifactId>
</dependency>
<dependency>
  <groupId>io.opentelemetry</groupId>
  <artifactId>opentelemetry-exporter-otlp</artifactId>
</dependency>
```

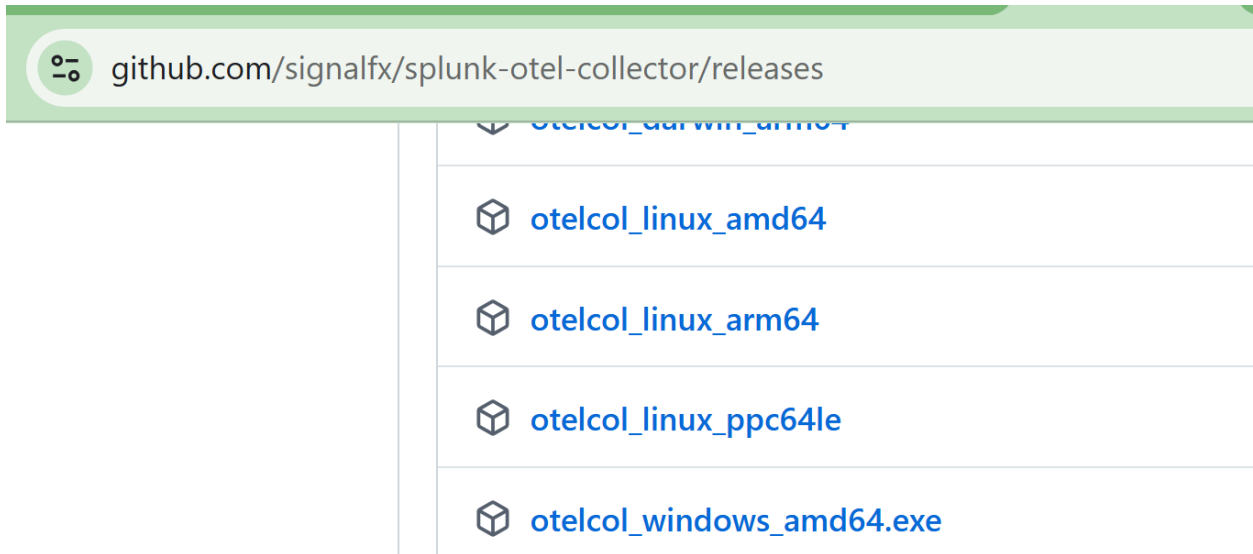
```
management:|
  tracing:
    sampling:
      probability: 1.0 # capture all traces (adjust in prod)
    otlp:
      tracing:
        endpoint: http://localhost:4317 # where your Splunk OTel Collector runs
```

```
spring:
  application:
    name: Microservice-3-CartService
  config:
    import: "optional:configserver:http://localhost:8880"
  logging:
  level:
  org:
    springframework:
    boot:
    context:
      config: TRACE
  cloud:
    config: DEBUG
management:
  tracing:
    sampling:
      probability: 1.0 # capture all traces (adjust in prod)
```

otlp:

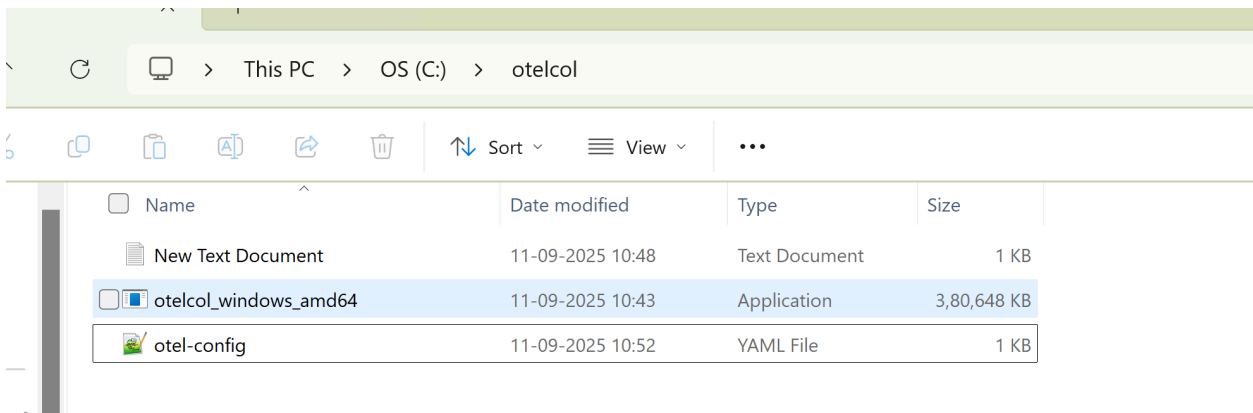
tracing:

endpoint: <http://localhost:4317> # where your Splunk OTel Collector runs



Otelcol\_windows\_amd64.exe install

```
PS C:\otelcol> .\otelcol_windows_amd64.exe --config .\otel-config.yaml
2025/09/11 10:53:28 settings.go:482: Set config to [.\otel-config.yaml]
2025/09/11 10:53:28 settings.go:543: Set memory limit to 460 MiB
2025/09/11 10:53:28 settings.go:528: Set soft memory limit set to 460 MiB
2025/09/11 10:53:28 settings.go:373: Set garbage collection target percentage (GOGC) to 400
```



```

PS C:\otelcol> .\otelcol_windows_amd64.exe --config .\otel-config.yaml
2025/09/11 10:55:05 settings.go:482: Set config to [.\otel-config.yaml]
2025/09/11 10:55:05 settings.go:543: Set memory limit to 460 MiB
2025/09/11 10:55:05 settings.go:528: Set soft memory limit set to 460 MiB
2025/09/11 10:55:05 settings.go:373: Set garbage collection target percentage (GOGC) to 400
2025/09/11 10:55:05 settings.go:418: set "SPLUNK_LISTEN_INTERFACE" to "0.0.0.0"
2025-09-11T10:55:05.219+0530 info service@v0.134.0/service.go:211 Starting otelcol... {"resource": {"service.i
instance.id": "55deafb6-a685-43d1-82e5-446111f867db", "service.name": "otelcol", "service.version": "v0.134.0"}, "Version
": "v0.134.0", "NumCPU": 8}
2025-09-11T10:55:05.220+0530 info extensions/extensions.go:41 Starting extensions... {"resource": {"service.i
instance.id": "55deafb6-a685-43d1-82e5-446111f867db", "service.name": "otelcol", "service.version": "v0.134.0"}}
2025-09-11T10:55:05.225+0530 info otlpreceiver@v0.134.0/otlp.go:121 Starting GRPC server {"resource": {"s
service.instance.id": "55deafb6-a685-43d1-82e5-446111f867db", "service.name": "otelcol", "service.version": "v0.134.0"},
"otelcol.component.id": "otlp", "otelcol.component.kind": "receiver", "endpoint": "127.0.0.1:4317"}
2025-09-11T10:55:05.227+0530 info otlpreceiver@v0.134.0/otlp.go:179 Starting HTTP server {"resource": {"s
service.instance.id": "55deafb6-a685-43d1-82e5-446111f867db", "service.name": "otelcol", "service.version": "v0.134.0"},
"otelcol.component.id": "otlp", "otelcol.component.kind": "receiver", "endpoint": "127.0.0.1:4318"}
2025-09-11T10:55:05.227+0530 info service@v0.134.0/service.go:234 Everything is ready. Begin running and processin
g data. {"resource": {"service.instance.id": "55deafb6-a685-43d1-82e5-446111f867db", "service.name": "otelcol", "service
.version": "v0.134.0"}}

```

receivers:

otlp:

protocols:

grpc:

http:

exporters:

splunkhec:

token: "8b0d5682-daf0-4fb4-ba62-c3278164f951"

endpoint: "https://localhost:8088/services/collector"

source: "otel"

sourcetype: "\_json"

index: "main"

tls:

insecure\_skip\_verify: true # 🖐 needed if Splunk HEC uses self-signed cert

service:

pipelines:

traces:

receivers: [otlp]

exporters: [splunkhec]

metrics:

receivers: [otlp]

exporters: [splunkhec]

logs:

receivers: [otlp]

exporters: [splunkhec]

```
> 9/11/25      { [-]
11:23:41.748 AM  logger: com.example.demo.controller.CartController
                  message: { [-]
                      endOfBatch: false
                      instant: { [+]
                          }
                      level: INFO
                      loggerFqn: org.apache.logging.log4j.spi.AbstractLogger
                      loggerName: com.example.demo.controller.CartController
                      message: CartController: Cart found for userId: 1
                      thread: http-nio-8083-exec-3
                      threadId: 112
                      threadPriority: 5
                  }
                  properties: { [-]
                      spanId: e1ce5871faed2746
                      traceId: a2e3e89f349ae988d6f3e89a65852599
                  }
                  severity: INFO
                  thread: http-nio-8083-exec-3
              }
```