spring-ai-alibaba-observability-example

Vulnerability Description:

POC：

Project information：

Vulnerability reproduction:

Code Analysis：

Vulnerability Description:

In the default configuration file, Springboot's Actuator is at risk of unauthorized access

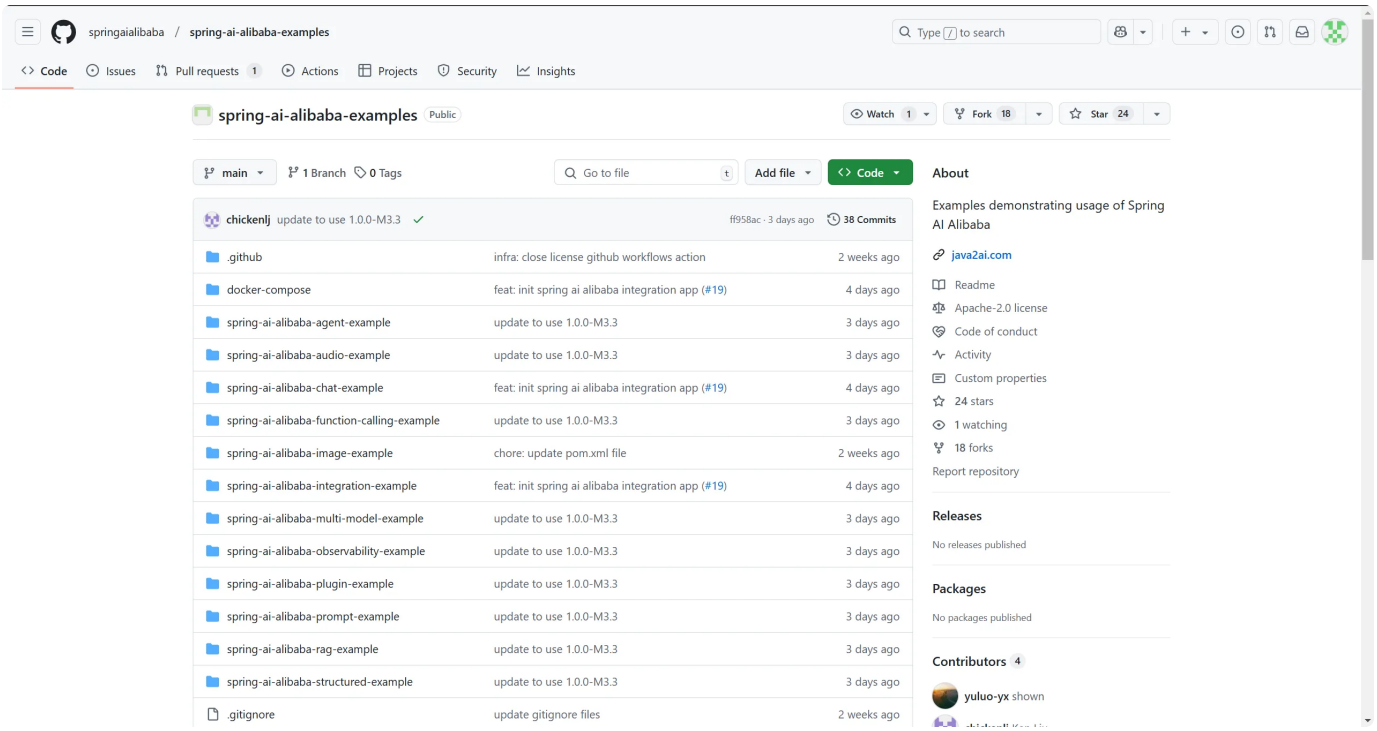
POC：

```java
#health or any other endpoint
curl http://localhost:8080/actuator/health
```

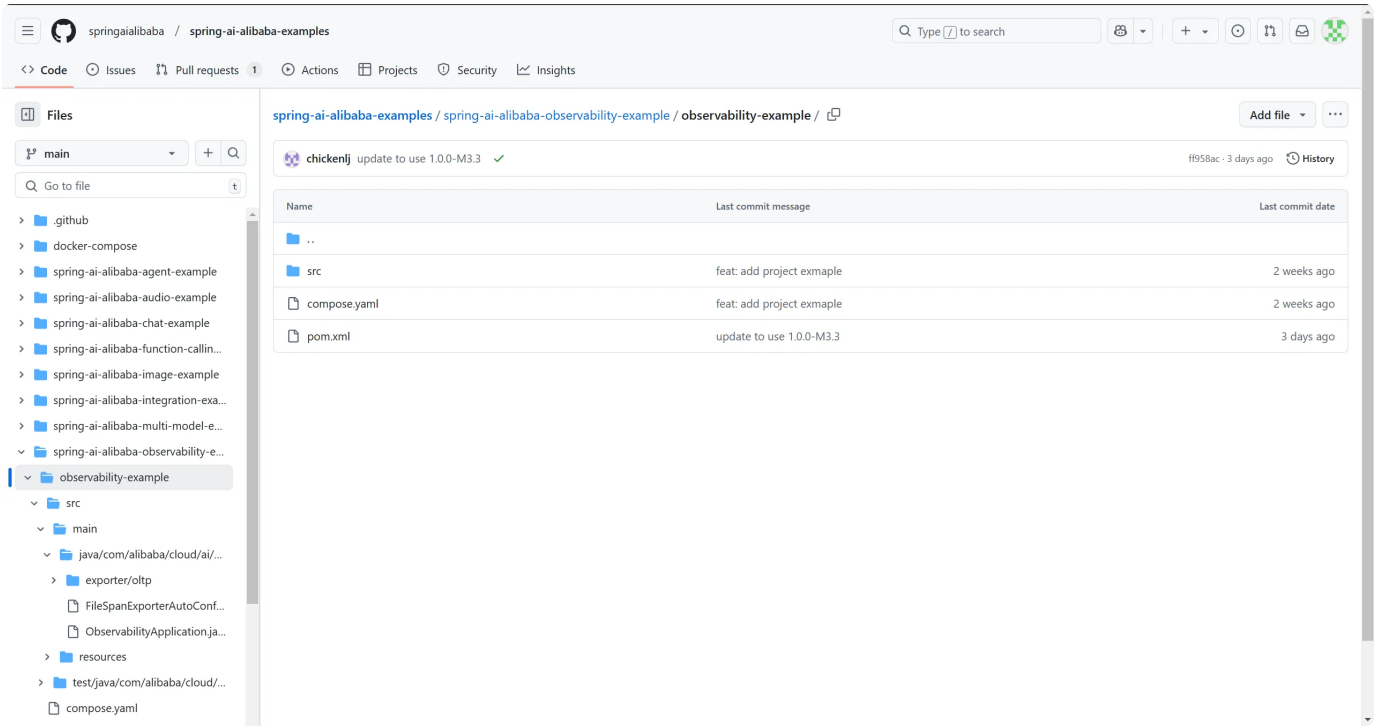
Project information：

Official website: java2ai.com

Project github: https://github.com/springaialibaba/spring-ai-alibaba-examples



And the location of the corresponding vulnerability code:
https://github.com/springaialibaba/spring-ai-alibaba-examples/tree/main/spring-ai-alibaba-observability-example/observability-example

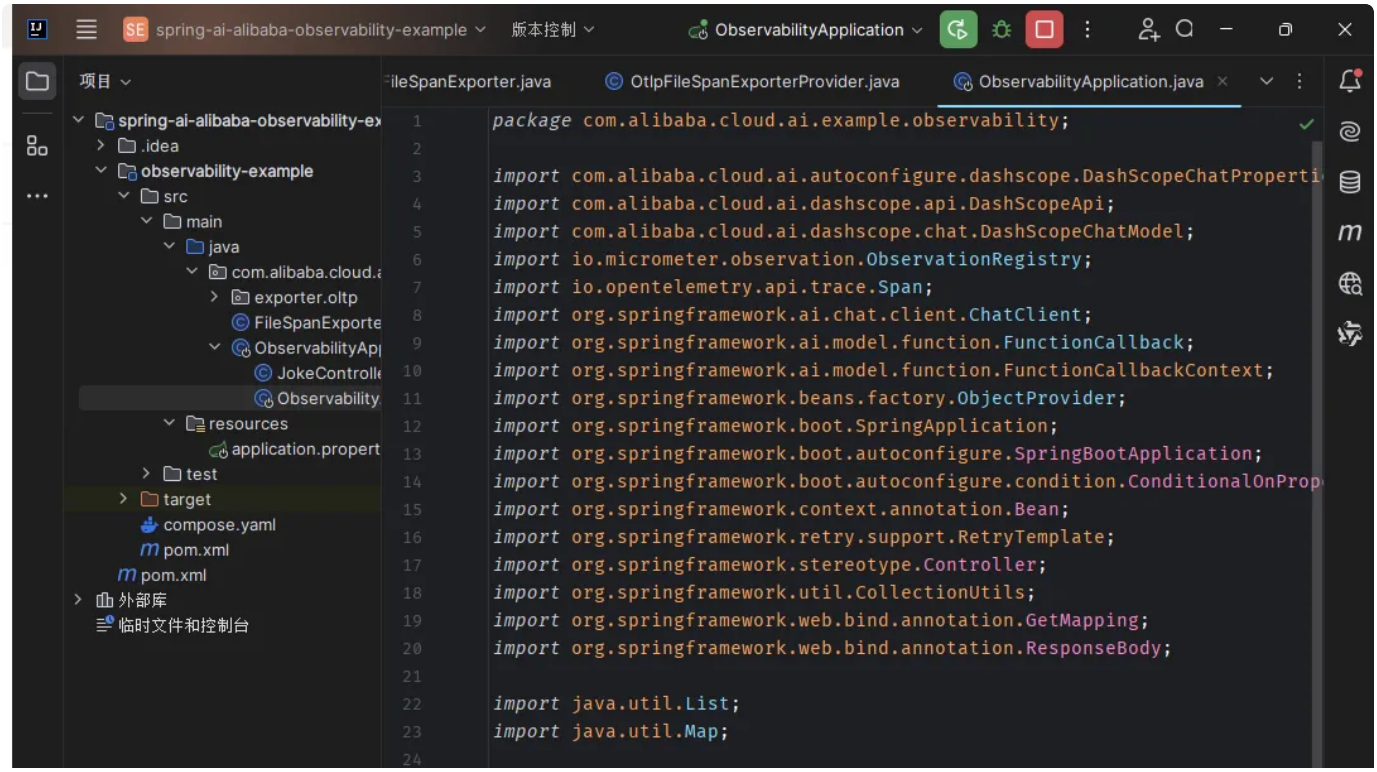


Vulnerability reproduction:

Add a new column for testing

```
C:\Users\[redacted]>curl http://localhost:8080/actuator/health
{"status":"DOWN","components":{"diskSpace":{"status":"UP","details":{"total":737277898752,"free":551480979456,"threshold
":10485760,"path":"E:\\development\\JAVA\\spring-ai-alibaba-examples-main\\spring-ai-alibaba-examples-main\\spring-ai-al
ibaba-observability-example\\.","exists":true}},"nacosConfig":{"status":"DOWN"},"ping":{"status":"UP"}}}
```

use the default code to start the application



```java
@SpringBootApplication
public class ObservabilityApplication {

    public static void main(String[] args) { SpringApplication.run(Observ

    @Bean
    ChatClient chatClient(ChatClient.Builder builder) { return builder.bu

    @Bean
    @ConditionalOnProperty(prefix = DashScopeChatProperties.CONFIG_PREFIX
            matchIfMissing = true)
    public DashScopeChatModel dashscopeChatModel(DashScopeChatProperties
                                                 FunctionCallbackContext
                                                 ObjectProvider<Observati

        if (!CollectionUtils.isEmpty(toolFunctionCallbacks)) {
            chatProperties.getOptions().getFunctionCallbacks().addAll(too
        }

        return new DashScopeChatModel(dashScopeApi, chatProperties.getOpt
                observationRegistry.getIfUnique(() -> ObservationRegistry
    }
}

@Controller
@ResponseBody
```

```
运行  ObservabilityApplication  ObservabilityApplication
控制台  Actuator
8080
.e.o.ObservabilityApplication     : Starting ObservabilityApplication using Java 17.0.13 with PID 45320 (
.e.o.ObservabilityApplication     : No active profile set, falling back to 1 default profile: "default"
Delegate$BeanPostProcessorChecker : Bean 'com.alibaba.cloud.nacos.NacosConfigAutoConfiguration' of type [
Delegate$BeanPostProcessorChecker : Bean 'nacosConfigProperties' of type [com.alibaba.cloud.nacos.NacosCo
.a.s.c.ClientAuthPluginManager    : [ClientAuthPluginManager] Load ClientAuthService com.alibaba.nacos.cl
.a.s.c.ClientAuthPluginManager    : [ClientAuthPluginManager] Load ClientAuthService com.alibaba.nacos.cl
Delegate$BeanPostProcessorChecker : Bean 'nacosConfigManager' of type [com.alibaba.cloud.nacos.NacosConfi
.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (http)
e.catalina.core.StandardService   : Starting service [Tomcat]
e.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.28]
.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
ervletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 869 ms
 potential conflicts
.ChatObservationAutoConfiguration : You have enabled the inclusion of the prompt content in the observati
.ChatObservationAutoConfiguration : You have enabled the inclusion of the completion content in the obser
StoreObservationAutoConfiguration : You have enabled the inclusion of the query response content in the o
.c.c.ChatClientAutoConfiguration  : You have enabled the inclusion of the input content in the observatio
ImageObservationAutoConfiguration : You have enabled the inclusion of the image prompt content in the obs
.a.NacosAnnotationProcessor       : [Nacos Config] Listening config for annotation: dataId=spring.ai.alib
.a.NacosAnnotationProcessor       : [Nacos Config] register configurablePromptTemplateFactory#method#onCo
.e.web.EndpointLinksResolver      : Exposing 16 endpoints beneath base path '/actuator'
.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) with context path '/'
com > alibaba > cloud > ai > example > observability > ObservabilityApplication.java > JokeController   62:1  LF  UTF-8  4 个空格
```

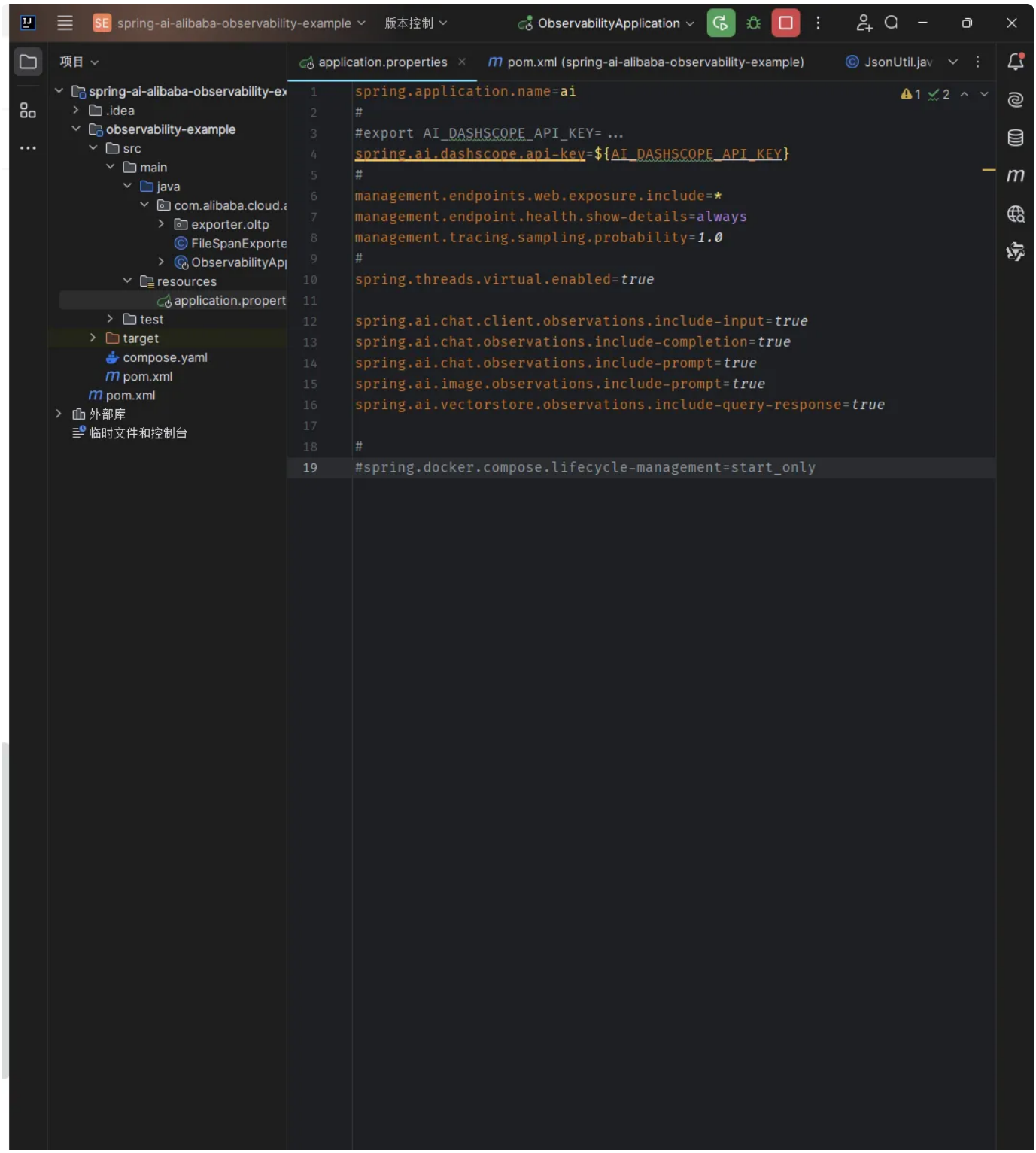
Then access the endpoint for testing, you can use the curl command or the browser to access it, and a valid health value will be returned.

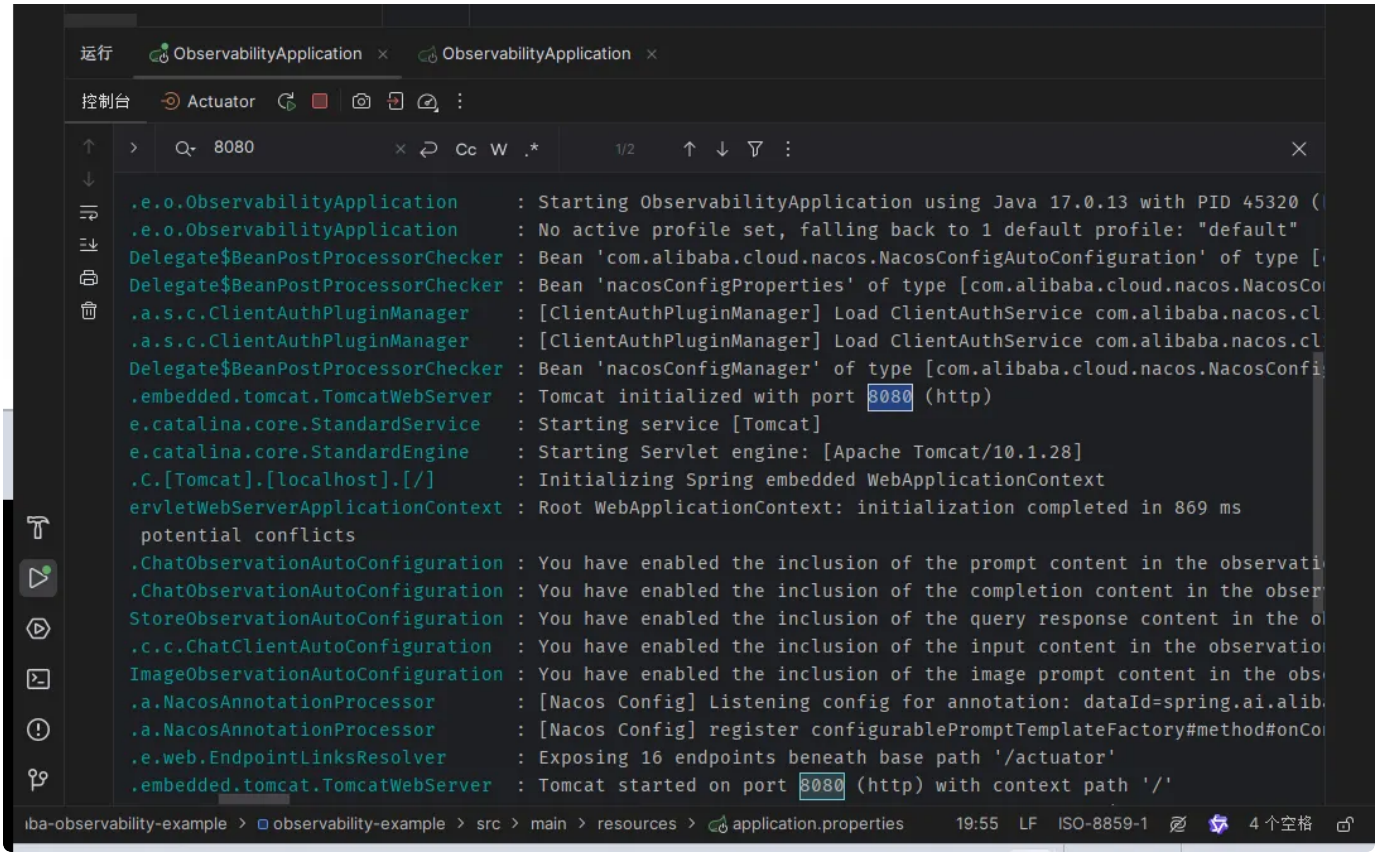
```
curl http://localhost:8080/actuator/health
```

```
C:\Users\[redacted]>curl http://localhost:8080/actuator/health
{"status":"DOWN","components":{"diskSpace":{"status":"UP","details":{"total":737277898752,"free":551480979456,"threshold
":10485760,"path":"E:\\development\\JAVA\\spring-ai-alibaba-examples-main\\spring-ai-alibaba-examples-main\\spring-ai-al
ibaba-observability-example\\.","exists":true}},"nacosConfig":{"status":"DOWN"},"ping":{"status":"UP"}}}
```

Code Analysis：

The configuration in this file exposes all Actuator endpoints and sets the health check endpoint to display detailed information:




运行
ObservabilityApplication
ObservabilityApplication
控制台
Actuator
8080
.e.o.ObservabilityApplication : Starting ObservabilityApplication using Java 17.0.13 with PID 45320 (
.e.o.ObservabilityApplication : No active profile set, falling back to 1 default profile: "default"
Delegate$BeanPostProcessorChecker : Bean 'com.alibaba.cloud.nacos.NacosConfigAutoConfiguration' of type [
Delegate$BeanPostProcessorChecker : Bean 'nacosConfigProperties' of type [com.alibaba.cloud.nacos.NacosCo
.a.s.c.ClientAuthPluginManager : [ClientAuthPluginManager] Load ClientAuthService com.alibaba.nacos.cl
.a.s.c.ClientAuthPluginManager : [ClientAuthPluginManager] Load ClientAuthService com.alibaba.nacos.cl
Delegate$BeanPostProcessorChecker : Bean 'nacosConfigManager' of type [com.alibaba.cloud.nacos.NacosConfi
.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)
e.catalina.core.StandardService : Starting service [Tomcat]
e.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.28]
.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
ervletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 869 ms
potential conflicts
.ChatObservationAutoConfiguration : You have enabled the inclusion of the prompt content in the observati
.ChatObservationAutoConfiguration : You have enabled the inclusion of the completion content in the obser
StoreObservationAutoConfiguration : You have enabled the inclusion of the query response content in the o
.c.c.ChatClientAutoConfiguration : You have enabled the inclusion of the input content in the observatio
ImageObservationAutoConfiguration : You have enabled the inclusion of the image prompt content in the obs
.a.NacosAnnotationProcessor : [Nacos Config] Listening config for annotation: dataId=spring.ai.alib
.a.NacosAnnotationProcessor : [Nacos Config] register configurablePromptTemplateFactory#method#onCo
.e.web.EndpointLinksResolver : Exposing 16 endpoints beneath base path '/actuator'
.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'
iba-observability-example > observability-example > src > main > resources > application.properties
19:55 LF ISO-8859-1 4 个空格