

Spring Boot Admin：微服务应用监控

Spring Boot Admin 可以对SpringBoot应用的各项指标进行监控，可以作为微服务架构中的监控中心来使用，本文将对其用法进行详细介绍。

Spring Boot Admin 简介

SpringBoot应用可以通过Actuator来暴露应用运行过程中的各项指标，Spring Boot Admin通过这些指标来监控SpringBoot应用，然后通过图形化界面呈现出来。Spring Boot Admin不仅可以监控单体应用，还可以和Spring Cloud的注册中心相结合来监控微服务应用。

Spring Boot Admin 可以提供应用的以下监控信息：

- 监控应用运行过程中的概览信息；
- 度量指标信息，比如JVM、Tomcat及进程信息；
- 环境变量信息，比如系统属性、系统环境变量以及应用配置信息；
- 查看所有创建的Bean信息；
- 查看应用中的所有配置信息；
- 查看应用运行日志信息；
- 查看JVM信息；
- 查看可以访问的Web端点；
- 查看HTTP跟踪信息。

创建admin-server模块

这里我们创建一个admin-server模块来作为监控中心演示其功能。

- 在pom.xml中添加相关依赖：

```
1 <dependency>
2     <groupId>org.springframework.boot</groupId>
3     <artifactId>spring-boot-starter-web</artifactId>
4 </dependency>
5 <dependency>
6     <groupId>de.codecentric</groupId>
7     <artifactId>spring-boot-admin-starter-server</artifactId>
8 </dependency>Copy to clipboardErrorCopied
```

- 在application.yml中进行配置：

```
1 spring:
2     application:
3         name: admin-server
4 server:
5     port: 9301Copy to clipboardErrorCopied
```

- 在启动类上添加@EnableAdminServer来启用admin-server功能：

```

1  @EnableAdminServer
2  @SpringBootApplication
3  public class AdminServerApplication {
4
5      public static void main(String[] args) {
6          SpringApplication.run(AdminServerApplication.class, args);
7      }
8
9  }Copy to clipboardErrorCopied

```

创建admin-client模块

这里我们创建一个admin-client模块作为客户端注册到admin-server。

- 在pom.xml中添加相关依赖：

```

1  <dependency>
2      <groupId>org.springframework.boot</groupId>
3      <artifactId>spring-boot-starter-web</artifactId>
4  </dependency>
5  <dependency>
6      <groupId>de.codecentric</groupId>
7      <artifactId>spring-boot-admin-starter-client</artifactId>
8  </dependency>Copy to clipboardErrorCopied

```

- 在application.yml中进行配置：

```

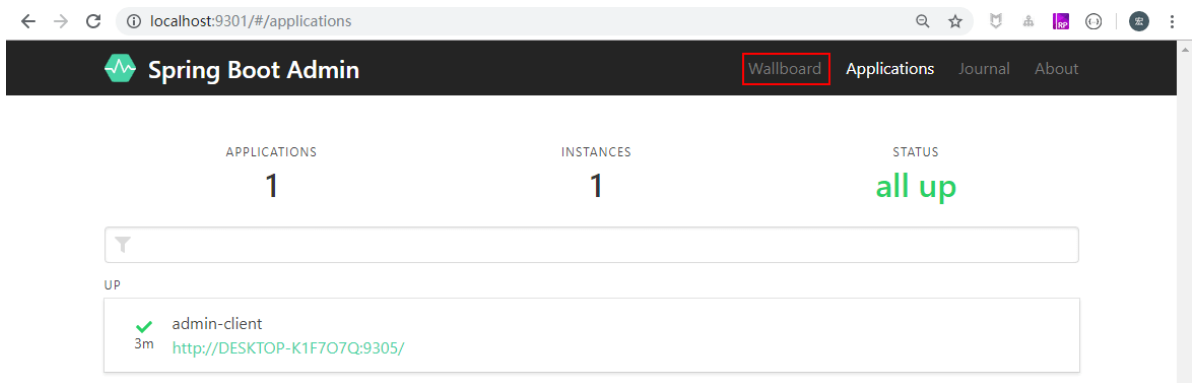
1  spring:
2      application:
3          name: admin-client
4      boot:
5          admin:
6              client:
7                  url: http://localhost:9301 #配置admin-server地址
8      server:
9          port: 9305
10     management:
11         endpoints:
12             web:
13                 exposure:
14                     include: '*'
15             endpoint:
16                 health:
17                     show-details: always
18     logging:
19         file: admin-client.log #添加开启admin的日志监控Copy to clipboardErrorCopied

```

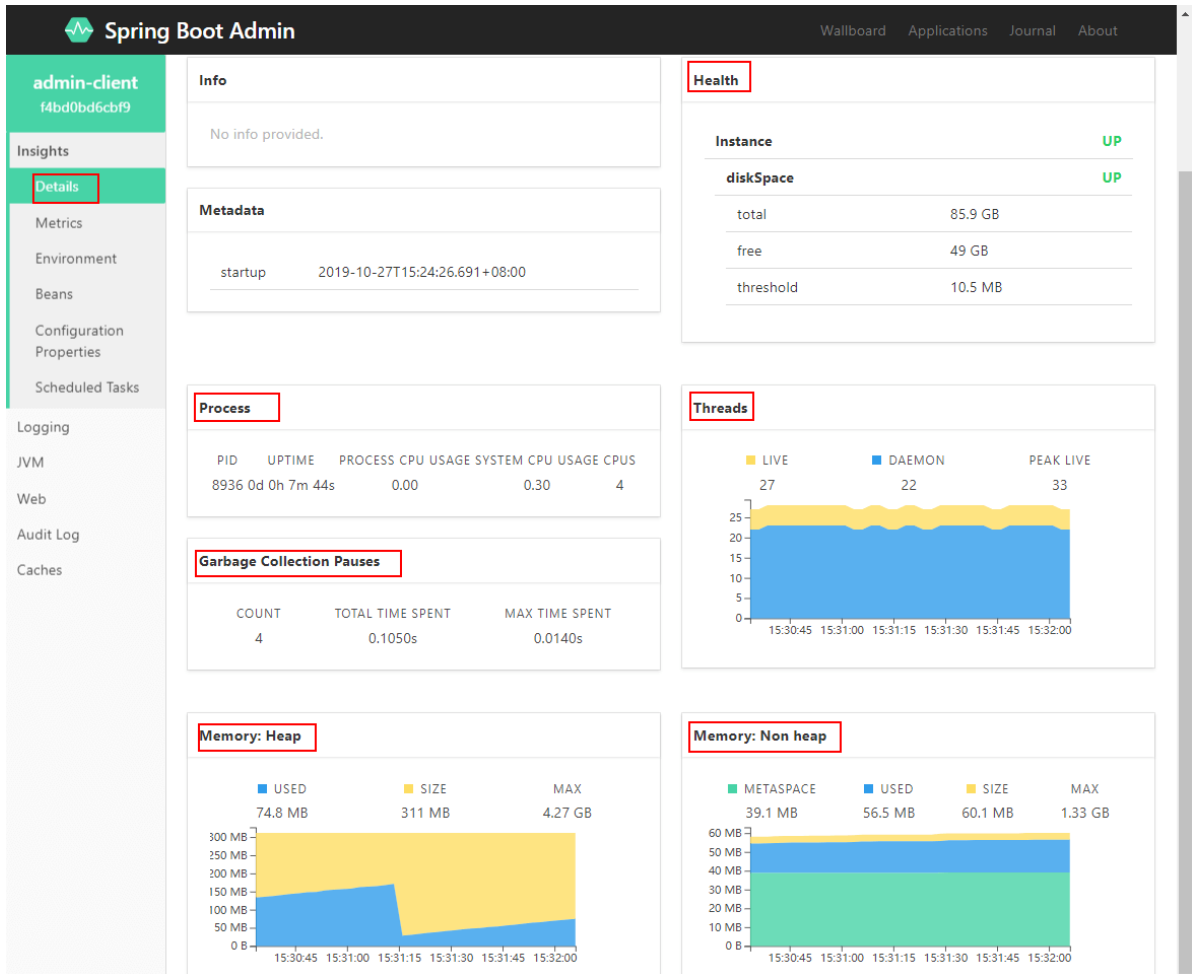
- 启动admin-server和admin-client服务。

监控信息演示

- 访问如下地址打开Spring Boot Admin的主页：<http://localhost:9301>



- 点击wallboard按钮，选择admin-client查看监控信息；
- 监控信息概览；



- 度量指标信息，比如JVM、Tomcat及进程信息；

Spring Boot Admin

Wallboard Applications Journal About

admin-client
f4bd0bd6cbf9

Insights

Details

Metrics

Environment

Beans

Configuration Properties

Scheduled Tasks

Logging

JVM

Web

Audit Log

Caches

jvm.memory.committed

http.server.requests
jvm.buffer.count
jvm.buffer.memory.used
jvm.buffer.total.capacity
jvm.classes.loaded
jvm.classes.unloaded
jvm.gc.live.data.size
jvm.gc.max.data.size
jvm.gc.memory.allocated
jvm.gc.memory.promoted
jvm.gc.pause
jvm.memory.committed
jvm.memory.max
jvm.memory.used
jvm.threads.daemon
jvm.threads.live
jvm.threads.peak
jvm.threads.states
logback.events
process.cpu.usage

VALUE

465895424

Add Metric

- 环境变量信息，比如系统属性、系统环境变量以及应用配置信息；

Spring Boot Admin

Wallboard Applications Journal About

admin-client
f4bd0bd6cbf9

Insights

Details

Metrics

Environment

Beans

Configuration Properties

Scheduled Tasks

Logging

JVM

Web

Audit Log

Caches

systemEnvironment

System Environment Property "=="

applicationConfig: [classpath:/application.yml]

spring.application.name	admin-client
class path resource [application.yml]:3:11	
spring.boot.admin.client.url	http://localhost:9301
class path resource [application.yml]:7:14	
server.port	9305
class path resource [application.yml]:9:9	
management.endpoints.web.exposure.include	*
class path resource [application.yml]:14:18	
management.endpoint.health.show-details	always
class path resource [application.yml]:17:21	
logging.file	admin-client.log
class path resource [application.yml]:19:9	

- 查看所有创建的Bean信息；

Spring Boot Admin

Wallboard Applications Journal About

admin-client
f4bd0bd6cbf9

usercontroller

admin-client

userController
com.macro.cloud.controller.UserController

singleton

Insights

Details

Metrics

Environment

Beans

Configuration Properties

Scheduled Tasks

Logging

JVM

Web

Audit Log

Caches

- 查看应用中的所有配置信息；

Spring Boot Admin

Wallboard Applications Journal About

admin-client
f4bd0bd6cbf9

server-org.springframework.boot.autoconfigure.web.ServerProperties

server.undertow.eagerFilterInit	true
server.undertow.accesslog.enabled	false
server.undertow.accesslog.pattern	common
server.undertow.accesslog.prefix	access_log.
server.undertow.accesslog.suffix	log
server.undertow.accesslog.dir	D:\developer\github\springcloud-learning\logs
server.undertow.accesslog.rotate	true
server.port	9305
server.maxHttpHeaderSize	{}
server.tomcat.accesslog.enabled	false
server.tomcat.accesslog.pattern	common
server.tomcat.accesslog.directory	logs
server.tomcat.accesslog.prefix	access_log
server.tomcat.accesslog.suffix	.log
server.tomcat.accesslog.rotate	true
server.tomcat.accesslog.renameOnRotate	false

Insights

Details

Metrics

Environment

Beans

Configuration Properties

Scheduled Tasks

Logging

JVM

Web

Audit Log

Caches

- 查看日志信息，需要添加以下配置才能开启；

```
1 logging:
2   file: admin-client.log #添加开启admin的日志监控Copy to clipboardErrorCopied
```

Spring Boot Admin

Wallboard Applications Journal About

admin-client f4bd0bd6cbf9

Insights

Logging

Logfile

Loggers

JVM

Web

Audit Log

Caches

2019-10-27 15:23:31.617 INFO 16180 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 9305 (http)

2019-10-27 15:23:31.645 INFO 16180 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2019-10-27 15:23:31.646 INFO 16180 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.22]

2019-10-27 15:23:31.754 INFO 16180 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2019-10-27 15:23:32.085 INFO 16180 --- [main] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 1323 ms

2019-10-27 15:23:32.274 INFO 16180 --- [main] o.s.b.a.e.web.servlet.endpoint.Registrar : Registered '/actuator/jolokia' to jolokia-actuator-endpoint

2019-10-27 15:23:32.544 INFO 16180 --- [main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 17 endpoint(s) beneath base path '/actuator'

2019-10-27 15:23:32.707 INFO 16180 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 9305 (http) with context path ''

2019-10-27 15:23:32.711 INFO 16180 --- [main] com.macro.cloud.AdminClientApplication : Started AdminClientApplication in 3.329 seconds (JVM running for 5.863)

2019-10-27 15:23:32.988 INFO 16180 --- [RMI TCP Connection(4)-127.0.0.1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'

2019-10-27 15:23:32.988 INFO 16180 --- [RMI TCP Connection(4)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'

2019-10-27 15:23:32.995 INFO 16180 --- [RMI TCP Connection(4)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Completed initialization in 7 ms

2019-10-27 15:24:23.608 INFO 8936 --- [main] com.macro.cloud.AdminClientApplication : Starting AdminClientApplication on DESKTOP-KIF707Q with PID 8936 (D:\dev\eloper\github\springcloud-learning\admin-client\target\classes started by zhenghong in D:\developer\github\springcloud-learning)

2019-10-27 15:24:23.612 INFO 8936 --- [main] com.macro.cloud.AdminClientApplication : No active profile set, falling back to default profiles: default

2019-10-27 15:24:25.714 INFO 8936 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 9305 (http)

2019-10-27 15:24:25.740 INFO 8936 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2019-10-27 15:24:25.741 INFO 8936 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.22]

2019-10-27 15:24:25.865 INFO 8936 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2019-10-27 15:24:25.866 INFO 8936 --- [main] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 2063 ms

2019-10-27 15:24:26.224 INFO 8936 --- [main] o.s.b.a.e.web.servlet.endpoint.Registrar : Registered '/actuator/jolokia' to jolokia-actuator-endpoint

2019-10-27 15:24:26.446 INFO 8936 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'

2019-10-27 15:24:26.740 INFO 8936 --- [main] o.s.s.c.ThreadPoolTaskScheduler : Initializing ExecutorService

2019-10-27 15:24:26.888 INFO 8936 --- [main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 17 endpoint(s) beneath base path '/actuator'

2019-10-27 15:24:26.969 INFO 8936 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 9305 (http) with context path ''

2019-10-27 15:24:26.973 INFO 8936 --- [main] com.macro.cloud.AdminClientApplication : Started AdminClientApplication in 4.655 seconds (JVM running for 5.715)

2019-10-27 15:24:27.323 INFO 8936 --- [RMI TCP Connection(2)-127.0.0.1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'

2019-10-27 15:24:27.324 INFO 8936 --- [RMI TCP Connection(2)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'

2019-10-27 15:24:27.334 INFO 8936 --- [RMI TCP Connection(2)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Completed initialization in 10 ms

2019-10-27 15:24:27.810 INFO 8936 --- [registrationTask1] d.c.b.a.c.r.ApplicationRegistrar : Application registered itself as f4bd0bd6cbf9

- 查看JVM信息;

Spring Boot Admin

Wallboard Applications Journal About

admin-client f4bd0bd6cbf9

Insights

Logging

JVM

JMX

Threads

Heap Dump

Web

Audit Log

Caches

MBeans

Connector 9305

Engine

J2EETYPE	WEBMODULE	NAME	J2EEAPPLICATION	J2EESERVER
Filter	//localhost/	Tomcat WebSocket (JSR356) Filter	none	none
Filter	//localhost/	characterEncodingFilter	none	none
Filter	//localhost/	formContentFilter	none	none
Filter	//localhost/	hiddenHttpMethodFilter	none	none
Filter	//localhost/	httpTraceFilter	none	none
Filter	//localhost/	requestContextFilter	none	none
Filter	//localhost/	webMvcMetricsFilter	none	none

DOMAINS

DefaultDomain

JMImplementation

Tomcat

com.sun.management

java.lang

java.nio

java.util.logging

jmx4perl

jolokia

org.springframework.boot

- 查看可以访问的Web端点;

Spring Boot Admin

Wallboard Applications Journal About

admin-client f4bd0bd6cbf9

Insights

Logging

JVM

Web

Mappings

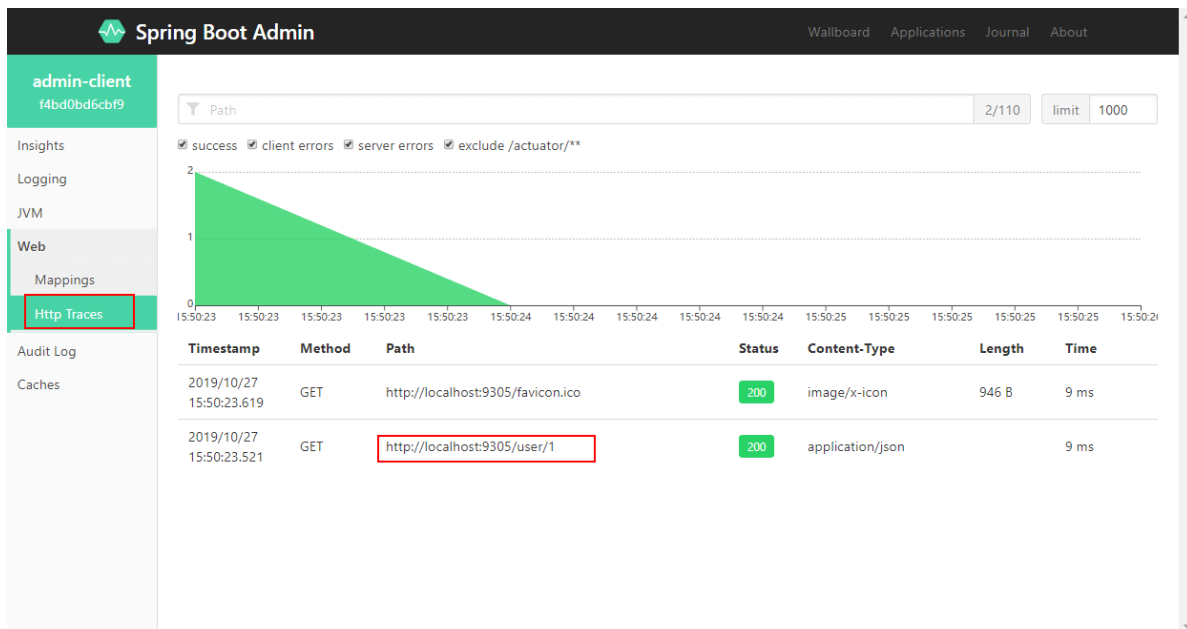
Http Traces

Audit Log

Caches

URI	Methods	Handler
/user/update	POST	public com.macro.cloud.domain.CommonResult com.macro.cloud.controller.UserController.update(com.macro.cloud.domain.User)
/user/delete/{id}	POST	public com.macro.cloud.domain.CommonResult com.macro.cloud.controller.UserController.delete(java.lang.Long)
/user/create	POST	public com.macro.cloud.domain.CommonResult com.macro.cloud.controller.UserController.create(com.macro.cloud.domain.User)
/user/getByUsername	GET	public com.macro.cloud.domain.CommonResult<com.macro.cloud.domain.User> com.macro.cloud.controller.UserController.getUsername(java.lang.String)
/user/getUserByIds	GET	public com.macro.cloud.domain.CommonResult<java.util.List<com.macro.cloud.domain.User>> com.macro.cloud.controller.UserController.getByIds(java.lang.String)

- 查看HTTP跟踪信息;



结合注册中心使用

Spring Boot Admin结合Spring Cloud 注册中心使用，只需将admin-server和注册中心整合即可，admin-server 会自动从注册中心获取服务列表，然后挨个获取监控信息。这里以Eureka注册中心为例来介绍下该功能。

修改admin-server

- 在pom.xml中添加相关依赖：

```
1 <dependency>
2   <groupId>org.springframework.cloud</groupId>
3   <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
4 </dependency>Copy to clipboardErrorCopied
```

- 在application-eureka.yml中进行配置，只需添加注册中心配置即可：

```
1 spring:
2   application:
3     name: admin-server
4 server:
5   port: 9301
6 eureka:
7   client:
8     register-with-eureka: true
9     fetch-registry: true
10    service-url:
11      defaultZone: http://localhost:8001/eureka/Copy to clipboardErrorCopied
```

- 在启动类上添加@EnableDiscoveryClient来启用服务注册功能：

```

1  @EnableDiscoveryClient
2  @EnableAdminServer
3  @SpringBootApplication
4  public class AdminServerApplication {
5
6      public static void main(String[] args) {
7          SpringApplication.run(AdminServerApplication.class, args);
8      }
9
10 }Copy to clipboardErrorCopied

```

修改admin-client

- 在pom.xml中添加相关依赖：

```

1  <dependency>
2      <groupId>org.springframework.cloud</groupId>
3      <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
4  </dependency>Copy to clipboardErrorCopied

```

- 在application-eureka.yml中进行配置，删除原来的admin-server地址配置，添加注册中心配置即可：

```

1  spring:
2      application:
3          name: admin-client
4  server:
5      port: 9305
6  management:
7      endpoints:
8          web:
9              exposure:
10                 include: '*'
11      endpoint:
12          health:
13              show-details: always
14  logging:
15      file: admin-client.log #添加开启admin的日志监控
16  eureka:
17      client:
18          register-with-eureka: true
19          fetch-registry: true
20          service-url:
21              defaultZone: http://localhost:8001/eureka/
Copy to clipboardErrorCopied

```

- 在启动类上添加@EnableDiscoveryClient来启用服务注册功能：

```

1  @EnableDiscoveryClient
2  @SpringBootApplication
3  public class AdminClientApplication {
4
5      public static void main(String[] args) {
6          SpringApplication.run(AdminClientApplication.class, args);
7      }
8
9  }Copy to clipboardErrorCopied

```


功能演示

- 启动eureka-server, 使用application-eureka.yml配置启动admin-server, admin-client;
- 查看注册中心发现服务均已注册: <http://localhost:8001/>

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
ADMIN-CLIENT	n/a (1)	(1)	UP (1) - DESKTOP-K1F7O7Q:admin-client:9305
ADMIN-SERVER	n/a (1)	(1)	UP (1) - DESKTOP-K1F7O7Q:admin-server:9301

- 查看Spring Boot Admin 主页发现可以看到服务信息: <http://localhost:9301>

The screenshot shows the Spring Boot Admin dashboard. At the top, there's a navigation bar with 'Spring Boot Admin' and links for 'Wallboard', 'Applications', 'Journal', and 'About'. Below the navigation bar, there are three summary cards: 'APPLICATIONS' with a value of '2', 'INSTANCES' with a value of '2', and 'STATUS' with the text 'all up'. Below these cards is a search bar and a list of instances. The list shows two instances: 'ADMIN-CLIENT' with a status of 'UP' and a URL 'http://DESKTOP-K1F7O7Q:9305/', and 'ADMIN-SERVER' with a status of 'UP' and a URL 'http://DESKTOP-K1F7O7Q:9301/'.

添加登录认证

我们可以通过给admin-server添加Spring Security支持来获得登录认证功能。

创建admin-security-server模块

- 在pom.xml中添加相关依赖:

```
1 <dependency>
2   <groupId>org.springframework.cloud</groupId>
3   <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>
4 </dependency>
5 <dependency>
6   <groupId>de.codecentric</groupId>
7   <artifactId>spring-boot-admin-starter-server</artifactId>
8   <version>2.1.5</version>
9 </dependency>
10 <dependency>
11   <groupId>org.springframework.boot</groupId>
12   <artifactId>spring-boot-starter-security</artifactId>
13 </dependency>
14 <dependency>
15   <groupId>org.springframework.boot</groupId>
16   <artifactId>spring-boot-starter-web</artifactId>
17 </dependency>
```

- 在application.yml中进行配置, 配置登录用户名和密码, 忽略admin-security-server的监控信息:

```
1 spring:
```

```

2     application:
3         name: admin-security-server
4     security: # 配置登录用户名和密码
5         user:
6             name: macro
7             password: 123456
8     boot: # 不显示admin-security-server的监控信息
9         admin:
10            discovery:
11                ignored-services: ${spring.application.name}
12     server:
13         port: 9301
14     eureka:
15         client:
16             register-with-eureka: true
17             fetch-registry: true
18             service-url:
19                 defaultZone: http://localhost:8001/eureka/

```

- 对SpringSecurity进行配置，以便admin-client可以注册：

```

1  /**
2   * Created by macro on 2019/9/30.
3   */
4  @Configuration
5  public class SecuritySecureConfig extends WebSecurityConfigurerAdapter {
6      private final String adminContextPath;
7
8      public SecuritySecureConfig(AdminServerProperties adminServerProperties) {
9          this.adminContextPath = adminServerProperties.getContextPath();
10     }
11
12     @Override
13     protected void configure(HttpSecurity http) throws Exception {
14         SavedRequestAwareAuthenticationSuccessHandler successHandler = new
15         SavedRequestAwareAuthenticationSuccessHandler();
16         successHandler.setTargetUrlParameter("redirectTo");
17         successHandler.setDefaultTargetUrl(adminContextPath + "/");
18
19         http.authorizeRequests()
20             //1.配置所有静态资源和登录页可以公开访问
21             .antMatchers(adminContextPath + "/assets/**").permitAll()
22             .antMatchers(adminContextPath + "/login").permitAll()
23             .anyRequest().authenticated()
24             .and()
25             //2.配置登录和登出路径
26             .formLogin().loginPage(adminContextPath +
27             "/login").successHandler(successHandler).and()
28             .logout().logoutUrl(adminContextPath + "/logout").and()
29             //3.开启http basic支持，admin-client注册时需要使用
30             .httpBasic().and()
31             .csrf()
32             //4.开启基于cookie的csrf保护
33             .csrfTokenRepository(CookieCsrfTokenRepository.withHttpOnlyFalse())
34             //5.忽略这些路径的csrf保护以便admin-client注册
35             .ignoringAntMatchers(
36                 adminContextPath + "/instances",

```

```

35         adminContextPath + "/actuator/**"
36     );
37 }
38 }Copy to clipboardErrorCopied

```

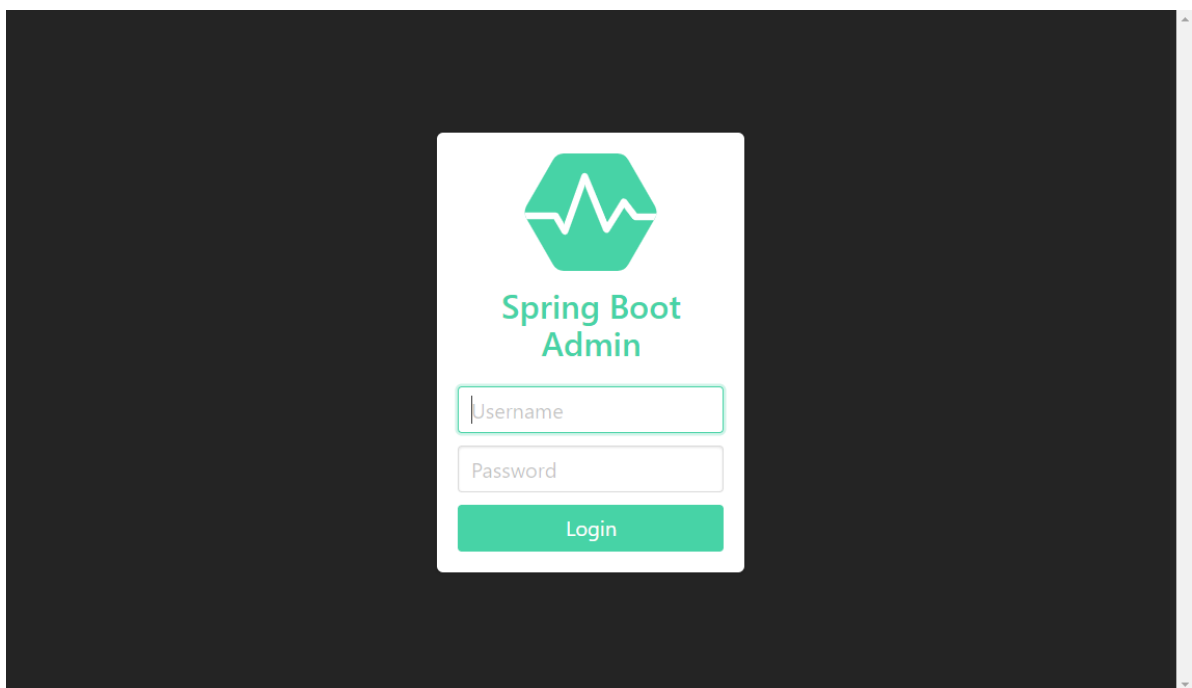
- 修改启动类，开启AdminServer及注册发现功能：

```

1  @EnableDiscoveryClient
2  @EnableAdminServer
3  @SpringBootApplication
4  public class AdminSecurityServerApplication {
5
6      public static void main(String[] args) {
7          SpringApplication.run(AdminSecurityServerApplication.class, args);
8      }
9
10 }Copy to clipboardErrorCopied

```

- 启动eureka-server，admin-security-server，访问Spring Boot Admin 主页发现需要登录才能访问：
<http://localhost:9301>



使用到的模块

```

1  springcloud-learning
2  ├── eureka-server -- eureka注册中心
3  ├── admin-server  -- admin监控中心服务
4  ├── admin-client  -- admin监控中心监控的应用服务
5  └── admin-security-server -- 带登录认证的admin监控中心服务

```