Web开发-JavaEE应用&SpringBoot栈&Actuator&Swagger&HeapDump&提取自动化



#知识点：  
1、安全开发-原生PHP-超级全局变量  
2、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
0、安全开发-原生PHP-数据库通讯  
1、安全开发-原生PHP-身份验证技术  
2、安全开发-Cookie&Session&Token  
3、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-弱类型脆弱  
2、安全开发-原生PHP-函数&数据类型  
3、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-开发组件集合  
2、安全开发-原生PHP-模版引擎渲染  
3、安全开发-原生PHP-第三方编辑器  
---------------------------------------------  
1、安全开发-框架技术-ThinkPHP  
2、安全开发-框架安全-版本&写法  
3、安全开发-ThinkPHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-文件安全操作  
2、安全开发-原生PHP-上传读取删除包含等  
3、安全开发-原生PHP-代码审计文件安全---------------------------------------------  
1、安全开发-原生PHP-RCE安全  
2、安全开发-原生PHP-代码执行&命令注入  
3、安全开发-原生PHP-代码审计案例  
---------------------------------------------  
1、安全开发-原生PHP-PHP.INI安全  
2、安全开发-原生PHP-全局文件&单函数  
3、安全开发-原生PHP-流量检测&AI算法  
---------------------------------------------  
1、安全开发-原生JS-语法模型概念  
2、安全开发-Ajax技术-JQuery&Axios  
3、安全开发-Ajax应用-文件上传&用户登录---------------------------------------------  
1、安全开发-原生JS-DOM树&BOM对象  
2、安全开发-原生JS-DOM安全&安全案例  
---------------------------------------------  
1、安全开发-原生JS-数据加密&代码混淆  
2、安全开发-原生JS-数据解密安全案例  
---------------------------------------------  
1、安全开发-NodeJS-开发环境&功能实现  
2、安全开发-NodeJS-安全漏洞&案例分析  
3、安全开发-NodeJS-特有漏洞&代码审计  
---------------------------------------------  
1、安全开发-WebPack-构建打包器  
2、安全开发-WebPack-源码泄漏还原  
---------------------------------------------  
1、安全开发-微信小程序-搭建&开发&架构&安全  
2、安全开发-微信小程序-编译调试&反编译&泄露  
---------------------------------------------  
1、安全开发-JavaEE-Servlet技术  
2、安全开发-JavaEE-监听器&过滤器  
---------------------------------------------  
0、安全开发-JavaEE-构建工具-Maven  
1、安全开发-JavaEE-ORM框架-JDBC  
2、安全开发-JavaEE-ORM框架-Mybatis  
3、安全开发-JavaEE-ORM框架-Hibernate  
4、安全开发-JavaEE-ORM框架-SQL注入&预编译  
---------------------------------------------  
1、安全开发-JavaEE-类加载器&反射机制&链安全  
2、安全开发-JavaEE-成员变量&成员方法&构造方法  
---------------------------------------------  
1、安全开发-JavaEE-动态代理&序列化&反序列化  
2、安全开发-JavaEE-readObject&toString方法---------------------------------------------  
1、安全开发-JavaEE-原生序列化-URLDNS链分析  
2、安全开发-JavaEE-FastJson-JdbcRowSetImpl链分析  
---------------------------------------------  
1、安全开发-JavaEE-JNDI注入-LADP&RMI&DNS等  
2、安全开发-JavaEE-JNDI注入-项目工具&手工原理等  
---------------------------------------------  
1、安全开发-JavaEE-第三方依赖开发安全  
2、安全开发-JavaEE-数据转换&FastJson&XStream  
3、安全开发-JavaEE-Shiro身份验证&Log4j日志处理  
---------------------------------------------  
1、安全开发-JavaEE-开发框架-SpringBoot&路由&传参  
2、安全开发-JavaEE-模版引擎-Thymeleaf&Freemarker&Velocity  
---------------------------------------------  
1、安全开发-JavaEE-常见依赖-Actuator&Swagger  
2、安全开发-JavaEE-安全问题-配置安全&接口测试  
   
#章节点：(待补充)  
#具体点：(待补充)



演示案例：

* WEB开发-JavaEE-Servlet&路由&生命周期
* WEB开发-JavaEE-过滤器Filter&监听器Listen
* WEB开发-JavaEE-Maven&JDBC&Mybatis&Hibernate
* WEB开发-JavaEE-ORM框架-SQL注入&预编译
* WEB开发-JavaEE-类加载器&反射机制&安全问题
* WEB开发-JavaEE-成员变量&构造方法&成员方法
* WEB开发-JavaEE-动态代理&序列化&反序列化
* WEB开发-JavaEE-利用链&危险方法&重写方法
* WEB开发-JavaEE-原生反序列化&URLDNS链
* WEB开发-JavaEE-FastJson反序列化&JDBC链
* WEB开发-JavaEE-JNDI注入&LDAP&RMI服务
* WEB开发-JavaEE-第三方依赖-FastJson&XStream
* WEB开发-JavaEE-第三方依赖-Shrio验证&Log4j日志
* WEB开发-JavaEE-开发框架-SpringBoot&路由&传参
* WEB开发-JavaEE-模版引擎-Thymeleaf&Freemarker&Velocity
* WEB开发-JavaEE-监控依赖-SpringBoot&Actuator&配置安全
* WEB开发-JavaEE-API依赖-SpringBoot&Swagger&接口测试

#开发框架-SpringBoot  
参考：<https://springdoc.cn/spring-boot/>  
   
#SpringBoot-监控依赖-Actuator  
SpringBoot Actuator模块提供了生产级别的功能，比如健康检查，审计，指标收集，HTTP跟踪等，帮助我们监控和管理Spring Boot应用。  
-开发使用：  
1、引入依赖  
<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-actuator</artifactId>  
</dependency>  
2、配置监控  
#暴露  
#基于application.properties  
management.endpoints.web.exposure.include=\*  
   
#基于application.yml  
management:  
 endpoints:  
 web:  
 exposure:  
 include: '\*'  
   
#安全配置：  
#application.properties  
management.endpoint.env.enabled=false  
management.endpoint.heapdump.enabled=false  
   
#application.yml  
management:  
 endpoint:  
 heapdump:  
 enabled: false #启用接口关闭  
 env:  
 enabled: false #启用接口关闭  
   
2、1 图像化Server&Client端界面  
Server：引入Server依赖-开启（@EnableAdminServer）  
Client：引入Client依赖-配置（连接目标，显示配置等）  
   
3、安全问题  
-heapdump泄漏  
jvisualvm分析器（自带）  
JDumpSpider提取器：<https://github.com/whwlsfb/JDumpSpider>  
heapdump\_tool提取器：<https://github.com/wyzxxz/heapdump_tool>  
分析提取出敏感信息（配置帐号密码，接口信息 数据库 短信 云应用等配置）  
   
4、额外安全：  
<https://github.com/LandGrey/SpringBootVulExploit>  
<https://github.com/wh1t3zer/SpringBootVul-GUI>  
例子：SpringCloud Gateway RCE（CVE-2022-22947）  
->创建SpringCloud Gateway+Actuator项目  
->更改项目版本及漏洞Gateway依赖版本  
<spring-boot.version>2.5.2</spring-boot.version>  
<spring-cloud.version>2020.0.3</spring-cloud.version>  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-gateway</artifactId>  
 <version>3.1.0</version>  
</dependency>  
->启动项目进行测试  
参考：https://www.cnblogs.com/qgg4588/p/18104875  
   
#SpringBoot-接口依赖-Swagger  
Swagger是当下比较流行的实时接口文文档生成工具。接口文档是当前前后端分离项目中必不可少的工具，在前后端开发之前，后端要先出接口文档，前端根据接口文档来进行项目的开发，双方开发结束后在进行联调测试。  
参考：<https://blog.csdn.net/lsqingfeng/article/details/123678701>  
-开发使用  
1、引入依赖  
<--2.9.2版本-->  
<dependency>  
 <groupId>io.springfox</groupId>  
 <artifactId>springfox-swagger2</artifactId>  
 <version>2.9.2</version>  
</dependency>  
<dependency>  
 <groupId>io.springfox</groupId>  
 <artifactId>springfox-swagger-ui</artifactId>  
 <version>2.9.2</version>  
</dependency>  
   
<--3.0.0版本-->  
<dependency>  
 <groupId>io.springfox</groupId>  
 <artifactId>springfox-boot-starter</artifactId>  
 <version>3.0.0</version>  
</dependency>  
   
2、配置访问  
#application.properties  
spring.mvc.pathmatch.matching-strategy=ant-path-matcher  
   
#application.yml  
spring  
 mvc:  
 pathmatch:  
 matching-strategy: ant\_path\_matcher  
   
2.X版本启动需要注释@EnableSwagger2  
3.X版本不需注释，写的话是@EnableOpenApi  
2.X访问路径：<http://ip:port/swagger-ui.html>  
3.X访问路径：http://ip:port/swagger-ui/index.html  
   
3、安全问题  
自动化测试：Apifox Reqable Postman  
泄漏应用接口：用户登录，信息显示，上传文件等  
可用于对未授权访问，信息泄漏，文件上传等安全漏洞的测试.

涉及资源：[资源下载地址](https://docs.qq.com/doc/DQ3Z6RkNpaUtMcEFr)