WebAssembly FaaS 平台
用户认证服务 调研(1)

2024年3月第五周 云原生技术组 组会

指导老师　孟宁　mengning@ustc.edu.cn
汇报人　戴明辰　daimingchen@mail.ustc.edu.cn

目录

1. 需求

- **独立的用户认证服务**
  - 身份认证（注册/登录）
    - 手机号为主(短信验证码登录)
    - 微信扫一扫快速登录(测试公众号账号)
  - 鉴权系统
    - Fine-grained permissions
  - 日志系统
    - Elasticsearch, Logstash, Kibana (ELK)?
    - Elasticsearch, Fluentd, Kibana (EFK)?
- 接口便于**集成**，使用方式**多样化**
  - Devstar console 端
  - Web端：Spring Boot + Vue 前后端分离
  - 桌面端：Tauri 框架
  - VS Code 插件

2. 案例

A. XXL-SSO
B. Apache Shiro
C. Keycloak

2.A 案例 XXL-SSO

- 分布式单点登录框架
- 优点
  - 支持Cookie和基于Token接入方式
  - 支持Web和APP接入，支持跨域应用
- 不足
  - 未实现API权限管理 (PermissionInterceptor没有实现)， 后期开发成本高
  - 项目停止维护多年
  - 开发思路过于简单，非业界认可的安全框架，安全性堪忧
    - 缺乏应对Web常见攻击：密码暴力破解、CSRF、XSS

2.B 案例 Apache Shiro (1) – 概览

- Apache Shiro 是 Apache软件基金会旗下的 Java 安全框架
- 优点
  - 支持**单点登录**
  - 支持**细粒度权限**管理
- 不足
  - 专用于Java语言的后端程序
  - 官方文档暂未提供 Kubernetes集群支持
    - 需要手工处理session共享相关逻辑
  - 缺乏 Security Threats Mitigation
    - 官方未应对Web常见攻击：密码暴力破解、CSRF、XSS

2.B 案例 Apache Shiro (2) - 核心概念

- Subject
  - the '*thing*' that is currently *interacting with the software*
    - a human being, a 3rd party process, a daemon account, etc.
- SecurityManager
  - manages security operations for *all* users
- Realms
  - configurations
    - authentication
    - authorization

2.B 案例 Apache Shiro (3) – 功能特性

```
1  @GetMapping("/config")
2  @RequiresPermissions("sys:oss:all")
3  public R config(){
4      CloudStorageConfig config = sysConfigService.getConfigObject(KEY,
5
6      return R.ok().put("config", config);
7  }
```

- Single Sign-On (SSO)
- Authentication (login)
- Authorization (access control)
  - **Roles**, e.g., “admin” , “teacher”, “student”
  - **Fine-grained permissions**, e.g., “sys:oss:all”, “course:list:write”
- Session Management
  - *Heterogeneous* client session access
    - Flash applets, C# applications, Java Web Start, and Web Applications, etc.

2.C 案例 Keycloak (1) – 概览

- Keycloak 是 *Red Hat* 开源的著名安全框架
  - Linux基金会项目
  - 云原生计算基金会(CNCF) 孵化项目
- 优点
  - 支持**单点登录**，支持持**细粒度权限**管理
  - 简单易用
    - 提供了 *Web* 管理界面
    - 提供了 *RESTful API* 接口
    - 适配多种客户端: *Spring, Tomcat, JavaScript, WildFly, JBoss EAP, Fuse, Jetty等*
  - 安全策略丰富，能抵御常见 Web攻击
- 不足
  - 官方已废弃Spring Boot适配器，需要手工集成

2.C 案例 Keycloak (2) – 核心概念

- Realms
  - 域
  - 管理着一批用户、证书、角色、组等（隔离）
- Clients
  - 客户端
  - 被KeyCloak保护的应用和服务
- Users
  - 用户
  - 使用登录系统
- Roles
  - 角色
  - 对用户的权限进行管理

2.C 案例 Keycloak (3) – 功能特性

- 拦截并跳转登录页面
  - 可自定义前端页面
    - 手机号注册功能
    - 手机号+密码/验证码登录
    - 微信扫一扫快速登录
  - 验证通过后重定向到受保护资源页面

2.C 案例 Keycloak (3) – 功能特性

2. 案例介绍 - 小结

- 我们需要什么
  - 单点登录功能
  - 利于多平台集成（Web端，桌面端，VS Code插件等）
    - 含有多种适配器，适配多种语言、不与任何语言强绑定
  - 安全
    - 广受业界认可的安全框架，并持续更新
    - 防止用户暴力破解，防止CSRF等Web常见攻击
- 结论：暂选 Keycloak

3. 规划

- Keycloak
  - 在Ubuntu VM 搭建 Keycloak k8s集群，配置数据库(MySQL or PostgreSQL)
  - 集成到分布式日志系统
    - Elasticsearch, Logstash, Kibana (ELK)?
    - Elasticsearch, Fluentd, Kibana (EFK)?
    - Even better than ELK/EFK?
  - 手工集成 Spring Boot，编写 demo
- Even better than Keycloak?
  - 继续寻找同类竞品

References

- XXL-SSO
  - https://www.xuxueli.com/xxl-sso
- Apache Shiro
  - https://shiro.apache.org/features.html
  - https://www.infoq.com/articles/apache-shiro/
- KeyCloak
  - https://www.keycloak.org/guides
  - https://www.keycloak.org/getting-started/getting-started-docker
  - https://www.keycloak.org/getting-started/getting-started-kube
  - https://www.keycloak.org/docs-api/24.0.1/javadocs/index.html
  - https://www.keycloak.org/docs-api/24.0.1/rest-api/index.html
  - https://blog.51cto.com/u_14230003/2511837
  - https://www.keycloak.org/server/logging