



Open Source > Development Lib > Reporting Tool

GVP anji-plus / AJ-Report

Watch 804 Star 7.5K

Code Issues 60 Pull Requests 2

Issues / 详情

Authentication Bypass vulnerability

Backlog #15VVZ0 JOHNSON Opened this issue 2022-10-15

这是英文的漏洞报告，中文的在(This is the English report, the Chinese report is in Chinese)

Description

The program uses a fixed JWT key, and the stored Redis key uses username in within an hour. JWT Token can be forged with his username to bypass

Login API

com.anjplus.template.gaea.business.modules.accessuser.controller.AccessUserController#login

```
93  /**
94   * 简单实现登录
95   * @param dto
96   * @return
97   */
98   @PostMapping("/{login}")
99   public ResponseBean login(@RequestBody @Validated GaeaUserDto dto) {
100       return responseSuccessWithData(accessUserService.login(dto));
101   }
```

Make redis key of format username, Although uuid is used, uuid is not involved in authentication.

com.anjplus.template.gaea.business.modules.accessuser.service.impl.AccessUserServiceImpl#login

com.anjplus.template.gaea.business.constant.BusinessConstant#GAEA_SECURITY_LOGIN_TOKEN

```
String loginName = gaeaUserDto.getLoginName();
String password = gaeaUserDto.getPassword();

// 1. 判断用户是否存在
LambdaQueryWrapper<AccessUser> wrapper = Wrappers.lambdaQuery();
wrapper.eq(AccessUser::getLoginName, loginName);
AccessUser accessUser = accessUserMapper.selectOne(wrapper);
if (null == accessUser) {
    throw BusinessExceptionBuilder.build(ResponseCode.LOGIN_ERROR);
}

// 2. 密码错误
if (!accessUser.getPassword().equals(MD5Util.encrypt(password))) {
    throw BusinessExceptionBuilder.build(ResponseCode.USER_PASSWORD_ERROR);
}

// 3. 如果该用户登录未过期，这里允许一个用户在多个终端登录
String tokenKey = String.format(BusinessConstant.GAEA_SECURITY_LOGIN_TOKEN, loginName);
String token = "";
GaeaUserDto gaeaUser = new GaeaUserDto();
if (cacheHelper.exists(tokenKey)) {
    token = cacheHelper.getString(tokenKey);
} else {
    // 生成用户token
    String uuid = GaeaUtils.UUID();
    token = jwtBean.createToken(loginName, uuid, type: 0, GaeaConstant.TENANT_CODE);
    cacheHelper.setString(tokenKey, token, seconds: 3600);
}
```

Uses a fixed JWT secret key

spring-boot-aaea-2.0.5.RELEASE.jar!com.anji.plus.aaea.utils.JwtBean#createToken

CLA

Gitee Pages JavaDoc sonarqube Quality Analysis

Jenkins for Gitee Baidu Efficiency Cloud Tencent CloudBase

Tencent Cloud Serverless 悬镜安全

Don't show this again

Status

Backlog

Assignees

Not set

Labels

Not set

Milestones

No related milestones

Pull Requests

None yet

Successfully merging a pull request.

Branches

No related branch

Planned to start - Planned to start

Unscheduled - Unschedule

Top level

Not Top

Priority

Not specified

参与者 (1)



[Explore](#)[Enterprise](#)[Education](#)[Gitee Premium](#)[Blog](#)[Go](#)

```

1 usage
2 Params: Username - 用户名
3 Returns:
4
5 1 usage
6
7 public String createToken(String username, String uuid, Integer type, String tenantCode) {
8     String token = JWT.create()
9         .withExpiresAt(GaeaDateUtils.toDate(LocalDate.now().plusHours(4)))
10        .withClaim(name: "username", username)
11        .withClaim(name: "uuid", uuid)
12        .withClaim(name: "type", type)
13        .withClaim(name: "tenant", tenantCode)
14        .sign(Algorithm.HMAC256(gaeaProperties.getSecurity().getJwtSecret()));
15    return token;
16 }
17
18 根据用户名、角色、权限、菜单等信息生成token

```



Gitee 已支持 CLA 协议签署

- 🔥 第一方功能集成，签署流程更高效
- 📄 内置可自定义的协议模板
- 👉 让开源贡献也能有据可依

[I know](#)[View Details](#)

TokenFilter for authentication

`com.anjplus.template.gaea.business.filter.TokenFilter#doFilter`

```

123 // 获取token
124 String token = request.getHeader("Authorization");
125 if (StringUtils.isBlank(token)) {
126     error(response);
127     return;
128 }
129
130 // 判断token是否过期
131 String loginName = jwtBean.getUsername(token);
132 String tokenKey = String.format(BusinessConstant.GAEA_SECURITY_LOGIN_TOKEN, loginName);
133 String userKey = String.format(BusinessConstant.GAEA_SECURITY_LOGIN_USER, loginName);
134 if (!cacheHelper.exists(tokenKey)) {
135     error(response);
136     return;
137 }
138
139 String gaeaUserJsonStr = cacheHelper.stringGet(userKey);
140
141 // 判断用户是否有该url的权限
142 if (!BusinessConstant.USER_ADMIN.equals(loginName)) {
143     AtomicBoolean authorizeFlag = authorize(request, gaeaUserJsonStr);
144     if (!authorizeFlag.get()) {
145         authError(response); // 无权限
146         return;
147     }
148 }

```

Forge different users' Tokens by modifying the username field

```

{
  "type": 0,
  "uuid": "",
  "tenant": "tenantCode",
  "username": "admin"
}

```

eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXB1IjowLCJ1dWkiOiJ0eXB1IiwidGvuwV50IjoidGvuwV50Q29kZSI6InVzZXJ1IiwiaWF0Ij0

The screenshot shows a Burp Suite interface with a 'Repeater' tab. A request is being sent to a web application, and the response is being decoded. The decoded response shows a JWT token. The token is being decoded using a tool like jwt.io, which shows the payload. The payload contains the username 'admin' and other metadata. The token is being used to forge different user tokens by modifying the username field.



Search...

0 matches

Search...

[+](#) JOHNSON created **任务** a month ago

[Sign in to comment](#)

[Contribution logs](#) [▼](#)



Gitee 已支持 CLA 协议签署

- 🔥 第三方功能集成，签署流程更高效
- 📄 内置可自定义的协议模板
- 👉 让开源贡献也能有据可依

[I know](#)

[View Details](#)



©OSCHINA. All rights reserved

[Git Resources](#)

[Learning Git](#)

[CopyCat](#)

[Downloads](#)

[Gitee Reward](#)

[Gitee Stars](#)

[Featured Projects](#)

[Blog](#)

[Nonprofit](#)

[Gitee Go](#)

[OpenAPI](#)

[Help Center](#)

[Self-services](#)

[Updates](#)

[About Us](#)

[Join us](#)

[Terms of use](#)

[Feedback](#)

[Partners](#)



[777320883](#)



git@oschina.cn



[Gitee](#)



[+86 400-606-0201](#)



Mini Program

