



链滴

关于 oAuth2 jwt authentication.getPrincipal 获取的是用户名的问题

作者: [Zcoin](#)

原文链接: <https://ld246.com/article/1595950894357>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

第一种方案重写转换类(DefaultUserAuthenticationConverter)

```
public Map<String, ?> convertUserAuthentication(Authentication authentication) {
    Map<String, Object> map = new HashMap<>(1);
    //继承UserDetails的对象
    Object o = authentication.getPrincipal();
    UserLogin userInfo = (UserLogin) o;
    //储存
    map.put("userInfo", userInfo);
    //权限存储
    if (authentication.getAuthorities() != null && authentication.getAuthorities().size()>0) {
        map.put(AUTHORITIES, AuthorityUtils.authorityListToSet(authentication.getAuthorities()));
    }
    return map;
}

public Authentication extractAuthentication(Map<String, ?> map) {
    Authentication authentication = null;
    if (map.containsKey("userInfo")) {
        Object principal = map.get("userInfo");
        Collection<? extends GrantedAuthority> authorities = this.getAuthorities(map);
        authentication = new UsernamePasswordAuthenticationToken(principal, "N/A", authorities);
    }
    return authentication;
}
```

第二种方案DefaultUserAuthenticationConverter设置customUserDetailsService

```
@Bean
public JwtAccessTokenConverter jwtAccessTokenConverter(){
    JwtAccessTokenConverter converter = new JwtAccessTokenConverter();
    //设置用户信息转换器，默认getPrincipal是username
    DefaultAccessTokenConverter accessTokenConverter = (DefaultAccessTokenConverter)converter.getAccessTokenConverter();
    DefaultUserAuthenticationConverter defaultUserAuthenticationConverter = new DefaultUserAuthenticationConverter();
    defaultUserAuthenticationConverter.setUserDetailsService(customUserDetailsService);
    accessTokenConverter.setUserTokenConverter(defaultUserAuthenticationConverter);
    ClassPathResource resource = new ClassPathResource("publicKey.txt");
    String publicKey = null;
    try {
        publicKey = IOUtils.toString(resource.getInputStream(), "UTF-8");
    } catch (IOException e) {
        e.printStackTrace();
    }
    converter.setVerifierKey(publicKey);
    return converter;
}
```

```
}
```

个人总结

//由于oauth2在jwt仅支持用户名和权限，如需扩展处理参考一下

//转换成token的核心代码，在DefaultUserAuthenticationConverter里最终调用
org.springframework.security.oauth2.provider.token.store.JwtAccessTokenConverter#convert
ccessToken

//解析token的核心代码，在DefaultUserAuthenticationConverter里最终调用
org.springframework.security.oauth2.provider.token.store.JwtAccessTokenConverter#extractA
thentication

//两者配合可以在jwt里面存放额外的用户信息(头像，卡号这些)
//但是我不推荐这么做，使用用户名单独查询用户信息更为合适。