**[#](https://pig4cloud.com/" \l "%E8%8E%B7%E5%8F%96%E5%BD%93%E5%89%8D%E7%94%A8%E6%88%B7) 获取当前用户**

SecurityUtils.getUser()

public PigUser getUser(Authentication authentication) {

Object principal = authentication.getPrincipal();

if (principal instanceof PigUser) {

return (PigUser) principal;

}

return null;

}

[**#**](https://pig4cloud.com/#%E4%B8%BA%E4%BB%80%E4%B9%88codengen-%E8%8E%B7%E5%8F%96%E7%94%A8%E6%88%B7%E4%B8%BA%E7%A9%BA) **为什么CodenGen 获取用户为空**

当在CodeGen模块，通过SecurityUtils.getUser() 的返回值始终为null,因为CodeGen重写了资源服务的配置,不通过pig获取用户信息提高性能  
ResourceServerConfigurer

@Configuration

@EnableResourceServer

@AllArgsConstructor

@EnableGlobalMethodSecurity(prePostEnabled = true)

public class ResourceServerConfigurer extends BaseResourceServerConfigurerAdapter {

/\*\*

\* 重写抽象类实现，不需要调用feign 获取 userDetailsService

\*

\* @param resources

\*/

@Override

public void configure(ResourceServerSecurityConfigurer resources) {

notGetUser(resources);

}

}

可以提供一个获取用户名的方法

public String getUsername(Authentication authentication) {

Object principal = authentication.getPrincipal();

return principal.toString();

}

资源服务器配置章节会详细讲重写和不重写两个的区别