1. 添加依赖

<!-- Authentication Attributes -->  
<dependency>  
 <groupId>org.apereo.cas</groupId>  
 <artifactId>cas-server-core-authentication-attributes</artifactId>  
 <version>${cas.version}</version>  
</dependency>

1. 修改 HTTPSandIMAPS-10000001.json 配置属性返回策略为默认（默认返回全部）

{  
 "@class" : "org.apereo.cas.services.RegexRegisteredService",  
 "serviceId" : "^(https|imaps|http)://.\*",  
 "name" : "HTTPS and IMAPS",  
 "id" : 10000001,  
 "description" : "This service definition authorizes all application urls that support HTTPS and IMAPS protocols.",  
 "evaluationOrder" : 10000,  
 "attributeReleasePolicy" : {  
 "@class" : "org.apereo.cas.services.ReturnAllAttributeReleasePolicy"  
 }  
}

1. 修改 application.properties

###单行  
# key为数据库的列，value为输出属性的key  
#columnMappings.attributes.[返回属性的key的键值]={查询单列得出来的字段}  
cas.authn.attributeRepository.jdbc[0].attributes.email=email  
#columnMappings.[返回属性的key的键值]={查询单列得出来的字段}  
cas.authn.attributeRepository.jdbc[0].columnMappings.email=email  
#确认输出单列  
cas.authn.attributeRepository.jdbc[0].singleRow=true  
#排序，默认为0  
cas.authn.attributeRepository.jdbc[0].order=0  
#数据库查询url,同上  
cas.authn.attributeRepository.jdbc[0].url=${cas.authn.jdbc.query[0].url}  
#查询where条件设定，由于登录查询用的关键字为telephone,登录之后到的是telephone,故规定username(我觉得不应该叫username，但官方规定)为telephone  
cas.authn.attributeRepository.jdbc[0].username=telephone  
cas.authn.attributeRepository.jdbc[0].user=${cas.authn.jdbc.query[0].user}  
cas.authn.attributeRepository.jdbc[0].password=${cas.authn.jdbc.query[0].password}  
cas.authn.attributeRepository.jdbc[0].sql=select \* from users where {0}  
cas.authn.attributeRepository.jdbc[0].dialect=${cas.authn.jdbc.query[0].dialect}  
cas.authn.attributeRepository.jdbc[0].ddlAuto=none  
cas.authn.attributeRepository.jdbc[0].driverClass=${cas.authn.jdbc.query[0].driverClass}  
cas.authn.attributeRepository.jdbc[0].leakThreshold=10  
cas.authn.attributeRepository.jdbc[0].propagationBehaviorName=PROPAGATION\_REQUIRED  
cas.authn.attributeRepository.jdbc[0].batchSize=1  
cas.authn.attributeRepository.jdbc[0].failFast=true