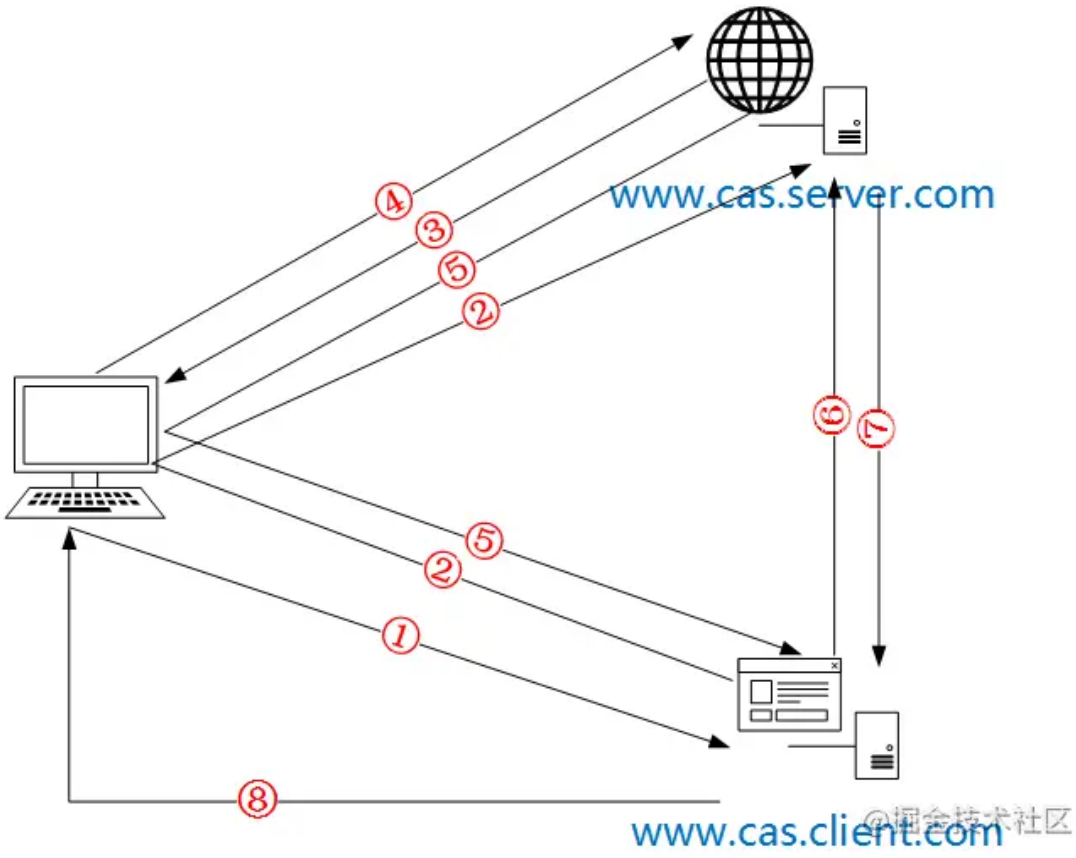
程序员Mark_Chou Lv2
2018年08月04日 阅读 7936

关注

终于搞明白了，CAS单点登录原理解析！！

真的如老话所说，“没有笨的人，只有懒得人”。前段时间时间需要和其他项目做cas集成，于是乎在网上找了几篇教程看了一下，好了，很简单，学会了，开搞（自以为研究明白）。集成完事了，登录成功了，自以为这就过去了。然而，没过几天就出bug了，这下惨了，当初没有好好学出了问题都不知道咋解决。无奈，只得静下心来好好学习一番（当初太懒付出的代价）。原理其实很简单的，只要耐下心来好好研究终会搞懂的。

先看下图



www.cas.client.com为cas客户端，也就是用户要访问的资源所在，www.cas.server.com为cas服务端，是单点登录的认证中心。

图中各步骤拆解说明：

则重定向到登录页面。

那么是如何验证用户是否登录过呢?

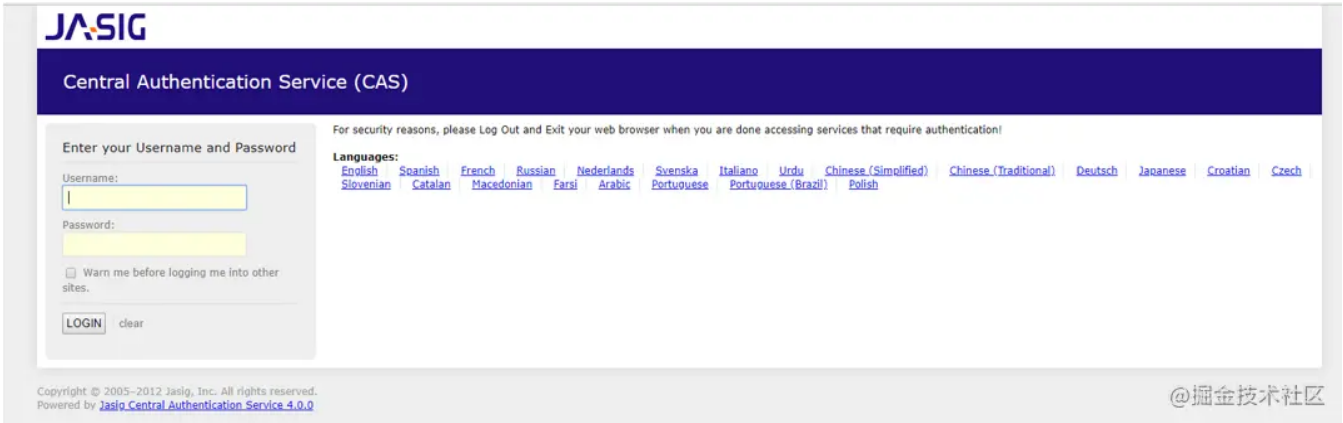
如果session中包含“const_cas_assertion”属性，说明已经登录，跳过此过滤器执行配置的其他过滤器；

如果ticket参数不为空（可能是登陆后跳转回来的），跳过此过滤器，执行TicketValidationFilter 验证ticket；

如果前两个条件都不满足，重定向到cas服务端，返回登录页面进行登录操作。

②：①中发现用户未登录，将浏览器重定向到www.cas.server.com，并携带一个参数service，参数值为①中的请求地址。

③：cas服务端收到请求将登录页面返回给浏览器。



④：用户输入用户名、密码，提交到cas服务端验证。

当cas服务端验证用户名、密码有效后，将浏览器重定向回①中service值对应的url并携带一个ticket参数，同时会在Cookie中设置一个CASTGC，该cookie是网站www.cas.server.com的cookie，只有访问这个网站才会携带这个cookie过去。

Cookie中的CASTGC：向cookie中添加该值的目的是当下次访问www.cas.server.com时，浏览器将Cookie中的TGC携带到服务器，服务器根据这个TGC，查找与之对应的TGT。从而判断用户是否登录过了，是否需要展示登录页面。TGT与TGC的关系就像SESSION与Cookie中SESSIONID的关系。

TGT：Ticket Granted Ticket（俗称大令牌，或者说票根，他可以签发ST）。

TGC：Ticket Granted Cookie（cookie中的value），存在Cookie中，根据他可以找到TGT。

ST：Service Ticket （小令牌），是TGT生成的，默认是用一次就生效了。也就是上面的ticket值。

⑥：www.cas.client.com取得ticket后进入TicketValidationFilter过滤器，该过滤器主要验证ticket是否有效。

⑧: www.cas.client.com将请求的资源返回给浏览器。

为了加深理解，也为了以后作参考，整理记录。另外，还要说一句，不要偷懒，多动手多动脑！

复制代码

```
参考文章

https://blog.csdn.net/javaloveiphone/article/details/52439613

https://www.cnblogs.com/lihuidu/p/6495247.html

https://blog.csdn.net/xiao__gui/article/details/38082761
```

*喜欢的小伙伴动动小手，点点关注。关注微信公众号【程序员Mark Chou 】,获取更多Java进阶、架构的干货资料（《Spring Cloud微服务实战》《Spring源码深度解析》《深入理解Apache Dubbo与实战》《一线架构师实践指南》《25大专题Java面试题手册》《Java面经》……）。

文章分类 后端 文章标签 后端