公众号: 后端元宇宙

目录导航

看完这篇你不能再说不懂SSO原理了！

这一篇是原理篇，接下来还会有一篇实战篇，实战的相关代码是非常火的一个开源项目叫:xxl-sso

一、简介

单点登录（Single Sign On），简称为 SSO。

它的解释是在多个应用系统中，用户只需要登录一次就可以访问所有相互信任的应用系统。

所谓一次登录，处处登录。同样一处退出，处处退出。

2024年3月(6)
2023年9月(4)
2023年6月(5)
2023年4月(3)
2023年2月(4)
2023年1月(3)
2022年11月(4)
2022年9月(4)
2022年6月(4)
2022年5月(5)
2022年3月(3)
2022年2月(1)
2022年1月(3)
2021年12月(6)
2021年11月(7)
更多

评论排行榜

1. 来博客园已过3年半了(57)
2. 高并发下秒杀商品，必须知道的9个细节(25)
3. Redisson实现分布式锁(1)---原理(24)
4. 算法(3)---布隆过滤器原理(23)
5. RocketMQ(1)-架构原理(19)
6. RocketMQ(2)---Docker部署RocketMQ集群(17)
7. Redisson实现分布式锁(3)—项目落地实现(16)
8. 分布式事务(4)---RocketMQ实现分布式事务项目(14)
9. 分布式事务(3)---RocketMQ实现分布式事务原理(14)
10. RocketMQ(5)---RocketMQ重试机制(14)
11. MySQL（12）---纪录一次left join一对多关系而引起的BUG(14)
12. 微信扫码登陆（2）---本地调试工具ngrok、微信回调ngrok域名(14)
13. SpringBoot(17）---SpringBoot整合RocketMQ(13)
14. Redisson实现分布式锁(2)—RedissonLock(12)

二、背景

[illegible]业内部使用的系统都会比较少，[illegible]有自己的登录功能。运营人员将自己的账号登录还是很方便。

但是随着公司的发展，公司的系统越来越多，比如有OA系统、CRM系统、财务管理系统、设备管理系统[illegible]个时候总不能每个系统都登录一遍吧，那真的会崩溃的。

合理做法是用户只需要登录一次就可以访问所有相互信任的应用系统。

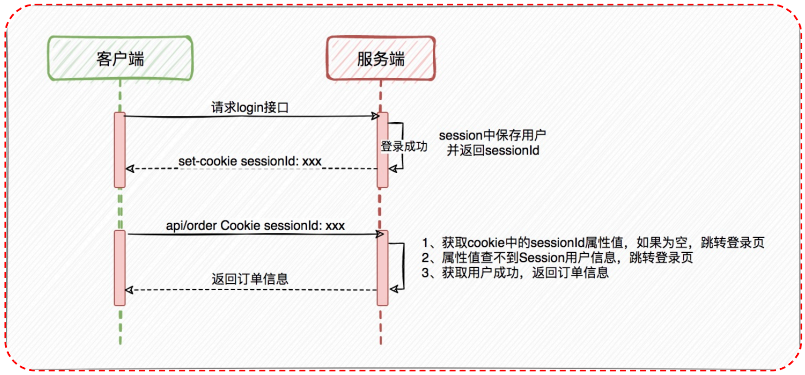
三、回顾下单系统登录是怎么样的?

我们都知道，http是无状态的协议,这意味着当你登录成功后请求其它接口服务端也并不知道你之前登录过。那怎么办呢?

这个时候我们会想到 Cookie + Session 组合来解决http无状态问题。

如果说 Cookie 是检查用户身上的"通行证"来确认用户的身份，那么 Session 就是通过检查服务器上的"客户明细表"来确认用户的身份的。

那这里完整的登录流程应该是这样的:



1)、 首次登录验证成功之后，后端会将用户信息存在Session对象中。

2)、同时设置 Set-Cookie 字段，并把 SessionId 等信息写入进去，并设置过期时间，这些信息就是 Cookie。浏览器会保存这些 Cookie 信息

3)、之后在请求该系统其它接口的时候,因为是同域名，浏览器会自动在请求头上添加 Cookie 字段，并带上保存的 Cookie 信

目录导航

15. SpringBoot(16)—@ConditionalOnBean与@ConditionalOnClass(11)

推荐排行榜

1. 【HTTP协议】---HTTP协议详解(81)
2. Redisson实现分布式锁(1)---原理(78)
3. RocketMQ(1)-架构原理(54)
4. 高并发下秒杀商品，必须知道的9个细节(53)
5. 分布式事务(1)---2PC和3PC原理(49)
6. 【TCP协议】(2)---TCP三次握手和四次挥手(47)
7. java代码之美（1）---Java8 Lambda(43)
8. 分库分表(1) --- 理论(37)
9. 【Git】(1)---工作区、暂存区、版本库、远程仓库(37)
10. SpringBoot(16)—@ConditionalOnBean与@ConditionalOnClass(35)

最新评论

1. Re:给你的 SpringBoot 工程部署的 jar 包瘦瘦身吧!
@愚夫c jdk必须一致，不然也容易出问题...
--景伟·郭
2. Re:数据库界的Swagger：一键生成数据库文档!
😃
--你会很厉害的
3. Re:Netty+WebSocket 获取火币交易所数据项目
现在这个火币的API连不上了
--sunny_HH
4. Re:给你的 SpringBoot 工程部署的 jar 包瘦瘦身吧!
写的详细，不错，但现在稍微规模一定的公司都是通过jekins这种持续集成工具来部署的，

息。

4)、后端接受到请求后,会在请求头中取出 SessionId的值，然
sion获取对于的用户信息，如果获取成功
不需要再重复登录。

总结 根据以上流程可知，SessionID 是连接 Cookie 和 Session 的一道桥梁，大部分系统也是根据此原理来验证用户登录状态。

所以，一般我们单系统实现登录会这样做：

- **登录**：将用户信息保存在Session对象中
-
  - 如果在Session对象中能查到，说明已经登录
  - 如果在Session对象中查不到，说明没登录（或者已经退出了登录）
- **注销（退出登录）**：从Session中删除用户的信息

四、多系统登录会存在的一些问题?

我们说单系统中登录流程实现关键点在于 Cookie 和 Session 的配合使用，但在多系统中就会存在很明显的两个问题

- 在多系统情况下服务端 Session 不共享。
- 在多系统情况下客户端 Cookie 不共享(跨域)。

如果能解决这两大难点，那实现多系统登录就简单多了

1.为什么会存在Session不共享?

我们说Session是存储在 服务端 的。

比如说现在有3台Tomcat服务器，当我们访问第1台Tomcat时，我们是可以将用户信息存在第1台Tomcat的Session中,但当我们访问第2台Tomcat的时候，这台服务器是

没有对应的Session数据,这就是所谓的Session不共享问题。

2、如何解决session共享问题呢?

说如何解决session共享问题呢，其实就是如何解决服务端数据共享问题

我们常见有3种解决方案：

第一种方案就是session拷贝。

你说的这种问题一般都不会太关注了

--kungge

5. Re:一文详解脏读、不可重复读、幻读

请问后面的文章发了吗？

--下沙grefus

6. Re:看一遍就懂：MVCC原理详解

MVCC能否解决了幻读问题那里说错了吧。 我查阅得知的是： 不可重复读是读取了其他事务更改的数据，针对update操作 幻读是读取了其他事务新增的数据，针对insert和delete操作 不可重复读（...

--zhu666

7. Re:order by 语句怎么优化?

如果select的字段比较多，这时候不能全加索引，所以避免不了走内部排序了吗，怎么优化呢

--pengfqa

8. Re:看完这篇你不能再说不懂SSO原理了!

全网把单点登录流程讲的最清楚的👍

--noodleOnce

9. Re:RocketMQ(5)---RocketMQ重试机制

那个timeout的例子我也没看懂，大佬能解释一遍吗？

--wsep

10. Re:docker-compose 搭建Prometheus+Grafana监控系统

还不错

--Edwin05

11. Re:OAuth 2.0详解

微信是服务商，A网站是第三方 第三方在登录时向服务商请求数据

--catcatcarrot

12. Re:看一遍就懂：MVCC原理详解

"不论是快照读和当前读都不能解决"这个说法有歧义，当前读其实就是加锁，也就是说当前

当某一台Tomcat对session中的信息进行了修改都会同步给其他Tomcat,这样session就可以共享。

储一份完整的session，增加服务器端压力也会浪费内存。

- 因为涉及到服务之间的同步，所以可能存在延迟。

第二种方案就是不通过session共享数据,而是采用redis。

redis纯天然解决了session不能共享的问题,而且redis除了存储查询效率高以外，还支持数据持久化功能，不用担心数据会丢失。

第二种方案也是现在现在企业级使用最多的一种方案。

第三种采用JWT。

我们在使用session或者使用redis,前端cookie其实只是存了个key，我们还需要拿着这个key到服务端的session，或者redis或者Mysql，总之都需要查一遍，但如果是JWT，

它最大的特点就是这个JWT本身就含有用户信息，服务端只要解析这个JWT成功，就可以获取用户信息。

3、为什么会Cookie跨域问题?

> 本质:由于浏览器安全策略，cookie只能在同一域名产生和使用

比方说，我们在请求www.a.com的时候，浏览器会自动把www.a.com的Cookie带去服务端。

但我们在请求www.b.com的时候，是不会把www.a.com下的Cookie带到b服务器的。

这就意味着由于域名不同，用户向系统A登录后，系统A返回给浏览器的Cookie，用户再请求系统B的时候不会将系统A的Cookie带过去。

至于如何解决Cookie跨域问题，不在这篇文章的讨论范畴内，下面实现单点登录的方式也不是通过解决Cookie跨域来实现的。

五、单点登录原理

目录导航

度其实可以解决MVCC的幻读问题

--iceqing

13. Re:给你的 SpringBoot 工程部署的 jar 包瘦瘦身吧!

java.lang.NoClassDefFoundError: org/springframework/boot/SpringApplication at com.dhpay.fundflow.App...

--一叶兰舟飘

14. Re:看完这篇你不能再说不懂SSO原理了!

mark

--大漠孤阳

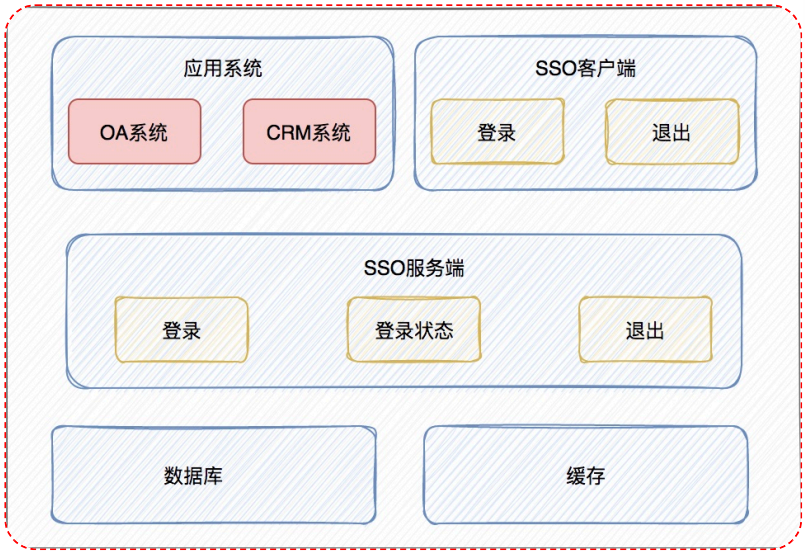
15. Re:Spring Event 观察者模式, 业务解耦神器

66666

--薛小谦

相比于单系统登录，sso需要一个独立的认证中心，只有认证中心能接受用户的用户名密码等安全信息，其他系统不提供登录入口，只接受认证中心的间接授权。

设计



应用系统：OA系统、CRM系统（需要登录的系统）

SSO客户端：登录、退出（独立jar包给应用系统引用）

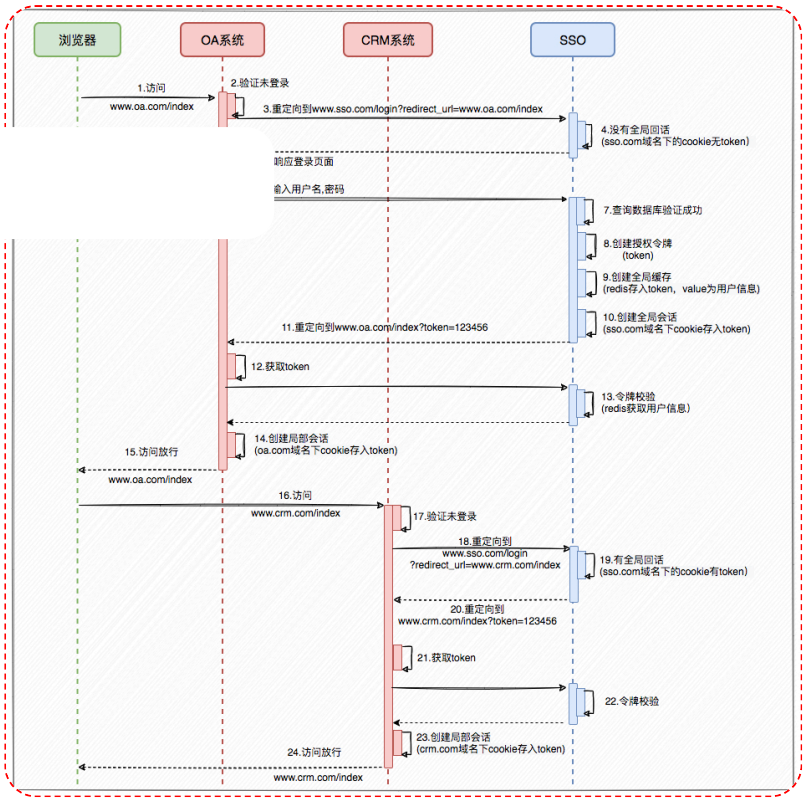
SSO服务端：登录（登录服务）、登录状态（提供登录状态校验/登录信息查询的服务）、退出（用户注销服务）

数据库：存储用户账户信息(一般使用Mysql)

缓存：存储用户的登录信息(一般使用Redis)

2、SSO登录流程

目录导航



对于这个流程图，我看网上问的最多的一个问题就是

> 根据同源策略:只要 协议+域名+端口号 一个不同，那么就不能进行跨域。www.oa.com 和 www.crm.com 域名都不相同了。也就是www.crm.com是拿不到www.oa.com中cookie中的token的,那crm.com在请求的时候为什么不需要登录呢?

其实这个问题，上面的流程图已经很清楚了。它也并不是通过解决跨域问题来实现单点登录的。

它实现的核心原理在于:

个人用户请求www.oa.com时，因为oa.com的cookie下没有token信息，所以跳转到sso.com/login,因为是第一次登录，所以sso.com的cookie下也没有token信息，所以需要

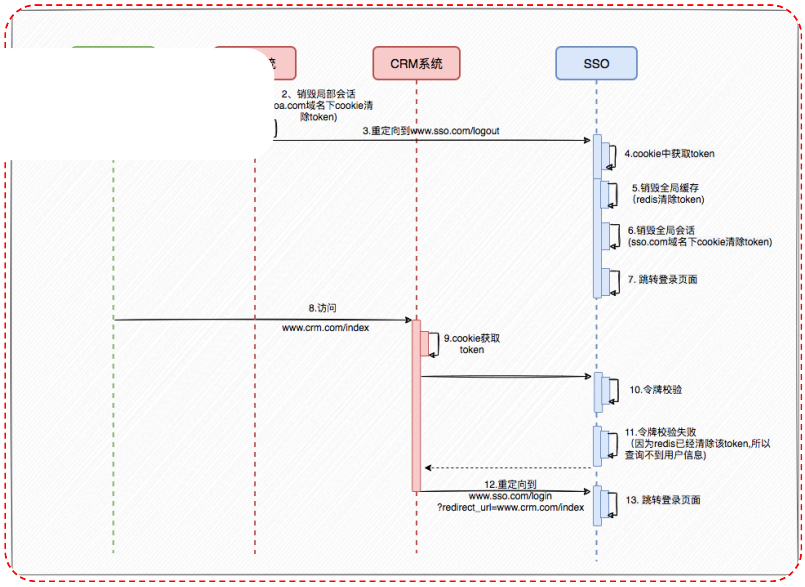
用户输入账号密码登录，登录成功会在sso.com域名下保存token信息，同时会把token信息返回给oa.com。

这样oa.com和sso.com下的cookie都有token信息。

而第一次访问crm.com的时候,它下面是没有token信息，所以会跳转到sso.com/login进行登录，但因为sso.com域名下cookie已经有token信息，所以不用再输入账号密码信息

直接把token返回到crm.com就可以，这个过程用户是无感知的，所以也就实现了一次登录处处登录了。

3、sso注销流程



对于这个流程，问的比较多的是：oa.com退出登录了。如何做到让crm.com也需要重新登录的?

通过上面的流程图我们可以知道www.oa.com退出登录，只能去除 oa.com 和 sso.com 域名下cookie下的token,但是crm.com域名下的cookie还是可以获取token的，

那能获取就代表这可以正常访问www.crm.com的接口了吗?

其实不是的，因为我们还有校验token有效性这一步(令牌校验)，我们拿着这个token去redis获取用户信息，其实已经获取不到了，因为上面退出登录的时候已经清除了，

所以令牌校验失败一样要重新登录。

声明：公众号如需转载该篇文章,发表文章的头部一定要 告知是转至公众号：后端元宇宙。同时也可以问本人要markdown原稿和原图片。其它情况一律禁止转载！

目录导航

分类： 【框架】-- SpringSecurity ， 【Study】--优化经验

好文要顶 关注我 收藏该文 微信分享

雨点的名字
粉丝 - 1528 关注 - 2
会员号：799
+加关注

« 上一篇： 浅谈微服务架构的演变史
» 下一篇： 推荐一个分布式单点登录框架XXL-SSO!

posted on 2023-02-17 09:12 雨点的名字 阅读(4200) 评论(5) 编辑 收藏 举报

会员力量，点亮园子希望 刷新页面 返回顶部

登录后才能查看或发表评论，立即 登录 或者 逛逛 博客园首页

目录导航

编辑推荐:

- ［架构师视角系列］风控场景下配置中心的设计实战
- 旧物利用 - 将机顶盒改造为一台 Linux 开发机!
- 这是DDD建模最难的部分（其实很简单）
- 为了落地DDD，我是这样“PUA”大家的
- 神秘 Arco 样式出现，祭出 Webpack 解决预期外的引用问题

阅读排行:

- 实习第一天，不小心透露了，我是拆迁户
- 记录兼职运维的一天
- 这就是为什么你学不会DDD
- 我们常用的地铁卡/银行卡，竟然运行着一个 Java 虚拟机
- 开源的 P2P 跨平台传文件应用「GitHub 热点速览」