

# 基于集群节点间即时拷贝的会话同步技术研究<sup>①</sup>

曹海涛, 胡 牧, 蒋厚明

(南瑞集团公司(国网电力科学研究院), 南京 210003)

**摘 要:** 集群会话同步技术目前主要有如下几种: 基于 Cookie 的会话同步, 基于数据库的会话同步, 基于分布式缓存的会话同步. 在以上会话同步技术的基础上, 提出了一种基于集群节点间即时拷贝的会话同步技术, 将会话标识信息存放在客户端中, 避免了在客户端中存放完整的会话信息, 从而可防止用户身份信息暴露的安全隐患; 同时, 在客户端向服务器发送请求的过程中, 只携带会话标识信息, 而不是完整的会话信息, 传输的数据量将大大减少, 提高了客户端对服务器的访问效率; 各集群节点间同步会话信息, 不需要从数据库中获取会话信息, 避免了频繁使用数据库带来的性能瓶颈; 也不需要专门使用专门的会话缓存服务器, 减低了开发和部署成本, 并具有良好的应用前景.

**关键词:** 集群服务; 会话; 会话同步; 即时拷贝

## Research on Session Synchronization Technology Based on Instant Copy of Cluster Node

CAO Hai-Tao, HU Mu, JIANG Hou-Ming

(Nari Group Corporation (State Grid Electric Power Research Institute), Nanjing 210003, China)

**Abstract:** Session synchronization technology mainly has the following several: session synchronization based on cookie, session synchronization based on database, session synchronization based on distributed cache. Based on the above session synchronization technologies, this paper gives a session synchronization technology based on instant copy of cluster node. This technology stores session id information in client to avoid store full session information in client, thus avoids the danger of user identity information exposure. Besides, in the process of sending a request to the server, only taking session id instead of complete session information. This greatly reduces transmitted data, so then improves the access efficiency of the client to server. Synchronizing session information between cluster nodes, there is no need to get session information from database which avoids performance bottlenecks on using database frequently. It also does not need specialized session cache servers, which cuts the cost of development and deployment. So, it has good application prospects.

**Key words:** cluster service; session; session synchronization; instant copy

## 1 引言

用户使用 Web 服务, 基本上需要客户端与 Web 服务器的多次交互. 而同一个用户和服务端的交互, 称为用户和服务器之间的会话. HTTP 协议本身是无状态的, 用户与 Web 应用交互过程信息(如购物车信息、用户状态、用户记录等)就只能依靠会话(session)技术<sup>[1]</sup>, 需要基于 HTTP 协议支持会话状态的机制. 而这样的机制能使 Web 服务器从多次单独的 HTTP 请

求中看到“会话”, 也就是知道请求是来自哪个会话的<sup>[2]</sup>. 具体实现方式为: 在会话开始时, 分配唯一的会话标识(SessionId), 通过某种手段把这个标识告诉客户端, 以后每次请求的时候, 客户端都会带上这个会话标识来告诉 Web 服务器请求是属于哪个会话的. Web 服务器集群是通过高速局域网互联的多台 Web 服务器组成的, 用户的 HTTP 请求被均衡地、透明地分配到集群中具体的服务器上, 由其完成请求响应过程, 并将响

<sup>①</sup> 收稿时间:2016-06-24;收到修改稿时间:2016-10-12 [doi: 10.15888/j.cnki.csa.005653]

应信息返回给用户<sup>[3]</sup>。假设来自同一客户端的两次请求被分发到了两台不同的 Web 服务器上,而这两次请求之间又存在关联(如:第一次请求是用户登录,第二次请求是获取信息),Web 服务器需要标识这两次请求是否来自同一个用户。此时就需要使用会话同步技术,会话同步技术是用来解决集群环境中,组成集群的各个 Web 服务器如何标识某一次客户端的请求是否属于同一会话的。本文所讨论的会话同步技术属于会话保持机制<sup>[4]</sup>的范畴。目前主流的会话同步技术主要有:基于 Cookie 的会话同步技术,基于数据库的会话同步技术和基于分布式缓存的会话同步技术。

在基于 Cookie 的会话同步方案中,服务端将会话信息返回给客户端,客户端将会话信息存储在 Cookie 中,客户端每次请求时携带 Cookie 中保存的会话信息,服务端根据 Cookie 中的会话信息创建会话。

在基于数据库的会话同步方案中,服务端将会话信息存放在数据库中,并将会话 ID 返回给客户端,客户端保存会话 ID 信息,客户端再次请求时,各集群节点根据客户端携带的会话 ID 从数据库中获取会话信息。

在基于分布式缓存的会话同步方案中,服务端将会话信息存放在分布式缓存服务器上,并将会话 ID 返回给客户端,客户端保存会话 ID 信息,客户端再次请求时,各集群节点根据会话 ID 从分布式缓存服务器上获取会话信息。

在此基础上,本文提出了一种基于集群节点间即时拷贝的会话同步技术,并将此技术应用于某企业移动应用项目的后台服务架构中,取得了良好的效果。

## 2 基于集群节点间即时拷贝的会话同步技术原理及实现

### 2.1 技术原理

基于集群节点间即时拷贝的会话同步技术实现原理如下:

(1) 客户端第一次访问集群服务时,当前服务的集群节点为客户端创建会话并保存在内存中,然后将会话标识反馈给客户端,客户端接收并保存会话标识,会话标识包括:会话 ID 和当前服务的集群节点 IP。

(2) 客户端再次访问集群服务时,需要携带会话标识,当前服务的集群节点根据客户端携带的会话标识解析出创建该会话的集群节点 IP 和会话 ID,并查看

本集群节点内是否存在该会话 ID,若存在,则根据该会话 ID 获取对应的会话信息;否则,根据创建该会话的集群节点 IP 和会话 ID 到对应的集群节点上获取会话信息。

(3) 某一集群节点中的会话信息更新时,此集群节点通知其他集群节点更新会话信息。

(4) 各集群节点定时清理超时的会话信息,以释放服务器的内存资源。

### 2.2 技术实现

基于集群节点间即时拷贝的会话同步技术具体实现步骤如图 1 所示。

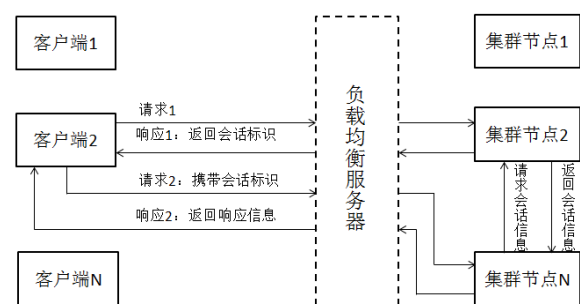


图 1 实施流程图

(1) 假设当前访问的客户端为客户端 2,客户端 2 第一次访问集群服务时,负载均衡服务将本次请求分发到集群节点 2 上,集群节点 2 将为客户端 2 创建会话信息生成会话标识,并将会话标识信息返回给客户端 2。

(2) 客户端 2 再次访问集群服务时,负载均衡服务将本次请求分发到集群节点 N 上(N 代表任意集群节点),集群节点 N 根据客户端 2 携带的会话标识信息,解析出集群节点 2 的 IP 和会话 ID,并根据会话 ID 在本节点中查找对应的会话信息,若找到,则将会话信息返回;否则,根据集群节点 2 的 IP 到集群节点 2 上获取会话 ID 对应的会话信息。获取会话信息后,根据客户端的会话 ID 校验客户端的请求是否合法,校验成功后,集群节点 N 将根据客户端的请求返回响应信息。

(3) 某个集群节点更新会话信息时,通知其他在线的集群节点更新会话信息,具体实现方式如下:更新会话信息的集群节点获取其他所有在线的集群节点,并远程调用各集群节点的更新会话接口,根据会话 ID 更新会话信息。

(4) 各集群节点定时清理无效的会话信息,以释

放无效的会话信息所占用的服务器内存资源。具体实现方式为：各集群节点遍历本节点内的会话信息，判断会话是否超时，若会话超时则清理。

### 3 各会话同步方案性能分析

基于 Cookie 的会话同步方案，客户端每次请求都需要携带完整的会话信息，当会话信息过大时，会导致请求到达服务端的时间过长，从而影响访问效率；并且，不同浏览器对 Cookie 使用有不同的限制，超过限制就会丢弃 Cookie，这会影晌应用系统的正常使用<sup>[5]</sup>。同时，由于每次请求都需要携带完整的会话信息，会增加暴露用户信息的风险。

基于数据库的会话同步方案，为同步会话信息需要频繁对数据库进行读写，在集群服务器较多，访问量较大时，容易造成性能瓶颈。

基于分布式缓存的会话同步方案，将会话信息存放在缓存服务器上，在系统部署时必须额外增加缓存服务器，所有对会话的操作都必须在缓存服务器上进行。在整个系统中使用会话缓存服务器将会增加系统部署和开发的难度，同时新增的会话缓存服务器将会带来新的维护成本。

基于集群节点间即时拷贝的会话同步方案，将会话信息存放在组成集群的各个节点服务器上，解决了基于分布式缓存的会话同步方案需要额外增加缓存服

务器的问题；将会话标识存放在客户端，解决了基于 Cookie 的会话同步方案将完整的会话信息存放在客户端的问题；同时，将会话信息存储在集群服务的各个节点的内存中，解决了基于数据库的会话同步方案中将会话信息存储在数据库中所带来的性能问题。

下表为各会话同步方案比较表，从表中可以看出，基于 Cookie 的会话同步方案，在客户端获取会话时，服务端的网络调用次数  $N$  和数据库访问次数  $D$  均为 0，但是该方案是将完整的会话信息存放在 Cookie 中，发送请求时需携带完整的会话信息，当会话信息较大时，必然会影晌访问效率，同时将会话信息存储在 Cookie 中，存在泄露会话信息的风险，安全性较低，而其他会话同步方案，将会话信息存储在服务端安全性相对较高。基于数据库的会话同步方案，在客户端获取会话时，服务端的网络调用次数  $N=0$ ，数据库访问次数  $D=1$ ，但是该方案中每次会话更新时都需要操作数据库，会造成性能瓶颈。基于分布式缓存的会话同步方案中， $N=1$ ，每次请求都需要从会话缓存服务器中获取会话信息；基于集群节点间即时拷贝的会话同步方案中， $N \leq 1$ ，只有本节点不存在会话信息时才会到其他节点同步会话信息，由此可以看出，在会话信息获取效率上，基于集群节点间即时拷贝的会话同步方案是较优的。

表 1 会话同步方案比较表

| 会话同步方案           | 服务端会话存储  | 客户端会话存储 | 会话信息安全 | 获取会话时，服务端网络调用次数 $N$ 和数据库访问次数 $D$ |
|------------------|----------|---------|--------|----------------------------------|
| 基于集群节点间即时拷贝的会话同步 | 集群节点内存中  | 会话 ID   | 高      | $N \leq 1, D=0$                  |
| 基于 Cookie 的会话同步  | 集群节点内存中  | 完整会话信息  | 低      | $N=0, D=0$                       |
| 基于数据库的会话同步       | 数据库中     | 会话 ID   | 高      | $N=0, D=1$                       |
| 基于分布式缓存的会话同步     | 会话缓存服务器中 | 会话 ID   | 高      | $N=1, D=0$                       |

### 4 结语

本文首先介绍客户端和 Web 服务器的会话以及会话同步的相关概念，指出在集群环境中会话同步的必要性，并介绍目前主流的会话同步技术，分析这些会话同步技术存在的问题。在此基础上提出一种新的会话同步技术：基于集群节点间即时拷贝的会话同步技术，重点探讨了基于集群节点间即时拷贝的会话同步技术的实现原理和具体实现步骤。

需要指出的是本文提出的会话同步技术的主要实现思路是在组成集群服务的各个集群节点间复制会话信息，以实现会话同步。该方案也存在一定局限性，在集群节点较多时，频繁地进行会话复制会带来较大的系统开销，影响系统的整体性能。在后续的工作中，作者将对此问题进行深入分析研究。

最后，笔者将该会话同步技术应用于某企业移动应用平台项目中，用于解决客户端移动设备和后台 Web 服务器集群之间的会话同步问题，取得了良好的效果。

### 参考文献

- 1 赵艳芳,蒋云起.Web 服务器负载均衡会话保持的研究.云南民族大学学报(自然科学版),2011,20(4):296-300.
- 2 曾宪杰.大型网站系统与 Java 中间件实践.北京:电子工业出版社,2014,(4):42-46.
- 3 郭成城,晏蒲柳.一种异构 Web 服务器集群动态负载均衡算法.计算机学报,2005,28(2):179-184.
- 4 Tsai HH, Sun DW. Color image watermark extraction based on support vector machines. Information Sciences, 2007, 177(2): 550-569.
- 5 许令波.深入分析 Java Web 技术内幕.北京:电子工业出版社, 2014.