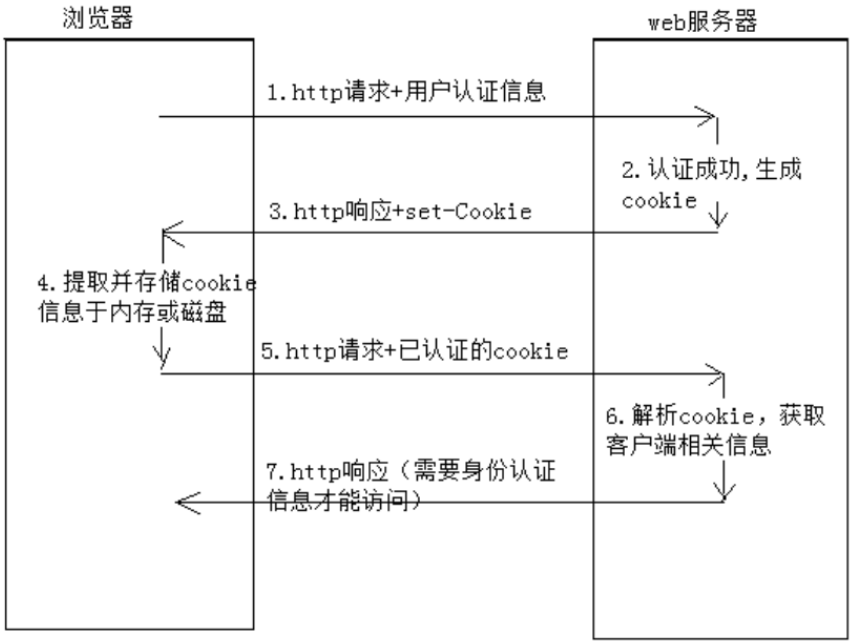
Cookie和Session

1 用户信息

`Http`是一个无状态协议, 就是说这一次请求和上一次请求是没有任何关系的，互不认识的，没有关联的。这种无状态的的好处是快速。坏处是需要进行用户状态保持的场景时[比如，登陆状态下进行页面跳转，或者用户信息多页面共享等场景]，必须使用一些方式或者手段比如：`session`和`cookie`

2 cookie

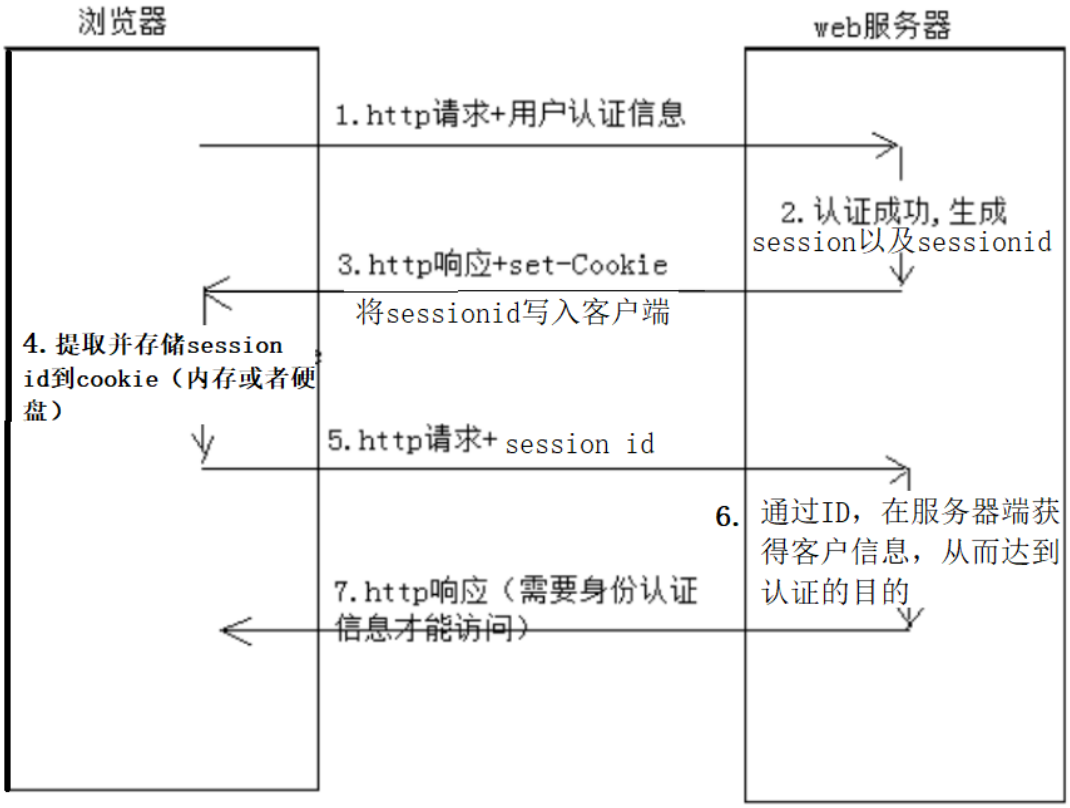
如上所述，`Http`是一个无状态的协议，但是访问有些资源的时候往往需要经过认证的账户才能访问，而且要一直保持在线状态，所以，cookie是一种在浏览器端解决的方案，将登陆认证之后的用户信息保存在本地浏览器中，后面每次发起http请求，都自动携带上该信息，就能达到认证用户，保持用户在线的作用，具体如下图：



设置cookie的方法在`Http`的`Response`报头中可以携带`Set-Cookie`字段来完成，后面会有演示。

3 session

而将用户敏感信息放到本地浏览器中，能解决一定的问题，但是又引进了新的安全问题，一旦cookie丢失，用户信息泄露，也很容易造成跨站攻击，所以有了另一种解决方法，将用户敏感信息保存至服务器，而服务器本身采用md5算法或相关算法生成唯一值（session id），将该值保存值客户端浏览器，随后，客户端的后续请求，浏览器都会自动携带该id，进而再在服务器端认证，进而达到状态保持的效果



4 cookie vs session

两者有什么区别呢?

- Cookie以文本文件格式存储在浏览器中，而session存储在服务端
- 因为每次发起`Http`请求，都要携带有效Cookie信息，所以Cookie一般都有大小限制，以防止增加网络压力,一般不超过4k
- 可以轻松访问cookie值但是我们无法轻松访问会话值，因此session方案更安全