

注意，所有匹配 cookie 都是和它们的 Set-Cookie2 过滤器一同传输的，而且保留关键字都是以美元符号 (\$) 开头的。

### 3. 版本1的Cookie2首部和版本协商

Cookie2 请求首部负责在能够理解不同 cookie 规范版本的客户端和服务端之间进行互操作性的协商。Cookie2 首部告知服务器，用户 Agent 代理理解新形式的 cookie，并提供了所支持的 cookie 标准版本（将其称为 Cookie-Version 更合适一些）：

```
Cookie2: $Version="1"
```

如果服务器理解新形式的 cookie，就能够识别出 Cookie2 首部，并在响应首部发送 Set-Cookie2（而不是 Set-Cookie）。如果客户端从同一个响应中既获得了 Set-Cookie 首部，又获得了 Set-Cookie2 首部，就会忽略老的 Set-Cookie 首部。

如果客户端既支持版本 0 又支持版本 1 的 cookie，但从服务器获得的是版本 0 的 Set-Cookie 首部，就应该带着版本 0 的 Cookie 首部发送 cookie。但客户端还应该发送 Cookie2: \$Version="1" 来告知服务器它是可以升级的。

## 11.6.8 cookie与会话跟踪

可以用 cookie 在用户与某个 Web 站点进行多项事务处理时对用户进行跟踪。电子商务 Web 站点用会话 cookie 在用户浏览时记录下用户的购物车信息。我们以流行的购物网站 Amazon.com 为例。在浏览器中输入 <http://www.amazon.com> 时，就启动了一个事务链，在这些事务中 Web 服务器会通过一系列的重定向、URL 重写以及 cookie 设置来附加标识信息。

图 11-5 显示了从一次 Amazon.com 访问中捕获的事务序列。

- 图 11-5a——浏览器首次请求 Amazon.com 根页面。
- 图 11-5b——服务器将客户端重定向到一个电子商务软件的 URL 上。
- 图 11-5c——客户端对重定向的 URL 发起一个请求。
- 图 11-5d——服务器在响应上贴上两个会话 cookie，并将用户重定向到另一个 URL，这样客户端就会用这些附加的 cookie 再次发出请求。这个新的 URL 是个胖 URL，也就是说有些状态嵌入到 URL 中去了。如果客户端禁止了 cookie，只要用户一直跟随着 Amazon.com 产生的胖 URL 链接，不离开网站，仍然可以实现一些基本的标识功能。
- 图 11-5e——客户端请求新的 URL，但现在会传送两个附加的 cookie。