13.5 安全性考虑

RFC 2617 总结了 HTTP 认证方案固有的一些安全风险，这是很令人钦佩的。本节描述了其中的部分风险。

13.5.1 首部篡改

为了提供一个简单明了的防首部篡改系统，要么就得进行端到端的加密，要么就得对首部进行数字签名——最好是两者的结合！摘要认证的重点在于提供一种防篡改认证机制，但并不一定要将这种保护扩展到数据上去。具有一定保护级别的首部只有 WWW-Authenticate 和 Authorization。

13.5.2 重放攻击

在当前的上下文中，重放攻击指的就是有人将从某个事务中窃取的认证证书用于另一个事务。尽管对 GET 请求来说这也是个问题，但为 POST 和 PUT 请求提供一种简单的方式来避免重放攻击才是非常必要的。在传输表单数据的同时，成功重放原先用过的证书会引发严重的安全问题。

因此，为了使服务器能够接受“重放的”证书，还必须重复发送随机数。缓解这个问题的方法之一就是让服务器产生的随机数包含（如前所述）根据客户端 IP 地址、时间戳、资源 Etag 和私有服务器密钥算出的摘要。这样，IP 地址和一个短小超时值的组合就会给攻击者造成很大的障碍。

但这种解决方案有一个很重要的缺陷。我们之前讨论过，用客户端 IP 地址来创建随机数会破坏经过代理集群的传输。在这类传输中。来自单个用户的多条请求可能会穿过不同的代理。而且，IP 欺骗也并不难实现。

一种可以完全避免重放攻击的方法就是为每个事务都使用一个唯一的随机数。在这种实现方式中，服务器会为每个事务发布唯一的随机数和一个超时值。发布的随机数只对指定的事务有效，而且只在超时值的持续区间内有效。这种方式会增加服务器的负担，但这种负担可忽略不计。

13.5.3 多重认证机制

服务器支持多重认证机制（比如基本认证和摘要认证）时，通常会在 WWW-Authenticate
303 首部提供选项。由于没有要求客户端选择功能最强的认证机制，所以得到的认证效果就和功能最弱的认证方案差不多。

要避免出现这个问题，最直接的方法就是让客户端总是去选择可用认证方案中功能