有时也将摘要函数称为加密的校验和、单向散列函数或指纹函数。

13.1.3 用随机数防止重放攻击

使用单向摘要就无需以明文形式发送密码了。可以只发送密码的摘要，而且可以确信，没有哪个恶意用户能轻易地从摘要中解码出原始密码。

但是，仅仅隐藏密码并不能避免危险，因为即便不知道密码，别有用心的人也可以截获摘要，并一遍遍地重放给服务器。摘要和密码一样好用。

为防止此类重放攻击的发生，服务器可以向客户端发送一个称为随机数（nonce）[9]的特殊令牌，这个数会经常发生变化（可能是每毫秒，或者是每次认证都变化）。客户端在计算摘要之前要先将这个随机数令牌附加到密码上去。 289

在密码中加入随机数就会使摘要随着随机数的每一次变化而变化。记录下的密码摘要只对特定的随机值有效，而没有密码的话，攻击者就无法计算出正确的摘要，这样就可以防止重放攻击的发生。

摘要认证要求使用随机数，因为这个小小的重放弱点会使未随机化的摘要认证变得和基本认证一样脆弱。随机数是在 `WWW-Authenticate` 质询中从服务器传送给客户端的。

13.1.4 摘要认证的握手机制

HTTP 摘要认证协议是一种升级版的认证方式，所用首部与基本认证类似。它在传统首部中添加了一些新的选项，还添加了一个新的可选首部 `Authorization-Info`。

图 13-2 描述了简化的摘要认证三步握手机制。

图 13-2 中发生的情况如下所述。

- 在第（1）步中，服务器会计算出一个随机数。在第（2）步中，服务器将这个随机数放在 `WWW-Authenticate` 质询报文中，与服务器所支持的算法列表一同发往客户端。 290
- 在第（3）步中，客户端选择一个算法，计算出密码和其他数据的摘要。在第（4）步中，将摘要放在一条 `Authorization` 报文中发回服务器。如果客户端要对服务器进行认证，可以发送客户端随机数。
- 在第（5）步中，服务器接收摘要、选中的算法以及支撑数据，计算出与客户端相同的摘要。然后服务器将本地生成的摘要与网络传送过来的摘要进行比较，验

注 9：随机数这个词表示“本次”或“临时的”。在计算机安全的概念中，随机数捕获了一个特定的时间点，将其加入到安全计算之中。