口令，如果口令正确则鉴别了 Alice。因为 Alice 不仅知道口令，而且知道用于加密口令的共享秘密密钥值，Bob 才可以轻松地鉴别 Alice 的身份。我们称这个协议为 ap3.1。

尽管协议 ap3.1 确实防止了 Trudy 得知 Alice 的口令，此处使用密码术并不能解决鉴别问题。Bob 受制于**回放攻击**（playback attack）：Trudy 只需窃听 Alice 的通信，并记录下该口令的加密版本，并向 Bob 回放该口令的加密版本，以假装她就是 Alice。协议 ap3.1 中加密口令的使用，并未使它比图 8-17 中的协议 ap3.0 的局面有明显改观。

8.4.5 鉴别协议 ap4.0

图 8-17 中的失败的情况是因为 Bob 不能区分 Alice 的初始鉴别报文和后来入侵者回放的 Alice 的初始鉴别报文所致。也就是说，Bob 无法判断 Alice 是否还活跃（即当前是否还在连接的另一端），或他接收到的报文是否就是前面鉴别 Alice 时录制的回放。观察力极强的读者会记起 TCP 的三次握手协议需要处理相同的问题，如果接收的 SYN 报文段来自较早连接的一个 SYN 报文段的旧副本（重新传输）的话，TCP 连接的服务器一侧不会接受该连接。TCP 服务器一侧如何解决“判断客户是否真正还活跃”的问题呢？它选择一个很长时间内都不会再次使用的初始序号，然后把这个序号发给客户，然后等待客户以包含这个序号的 ACK 报文段来响应。此处我们能够为鉴别目的采用同样的思路。

不重数（nonce）是在一个协议的生存期中只使用一次的数。也就是说，一旦某协议使用了一个不重数，就永远不会再使用那个数字了。协议 ap4.0 以如下方式使用一个不重数：

1）Alice 向 Bob 发送报文“我是 Alice”。

2）Bob 选择一个不重数 R，然后把这个值发送给 Alice。

3）Alice 使用她与 Bob 共享的对称秘密密钥 K_{A-B}来加密这个不重数，然后把加密的不重数 $K_{A-B}(R)$ 发回给 Bob。与在协议 ap3.1 中一样，由于 Alice 知道 K_{A-B}并用它加密一个值，就使得 Bob 知道收到的报文是由 Alice 产生的。这个不重数用于确定 Alice 是活跃的。

4）Bob 解密接收到的报文。如果解密得到的不重数等于他发送给 Alice 的那个不重数，则可鉴别 Alice 的身份。

协议 ap4.0 如图 8-18 所示。通过使用这个在生存期中只出现一次的值 R，然后核对返回的值 $K_{A-B}(R)$，Bob 能够确定两点：Alice 是她所声称的那个人（因为她知道加密 R 所需的秘密密钥），Alice 是活跃的（因为她已经加密了 Bob 刚刚产生的不重数 R）。

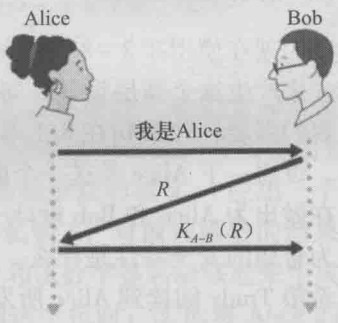

图 8-18 协议 ap4.0：无失败的情况

不重数和对称密钥密码体制的使用形成了 ap4.0 的基础。一个自然的问题是，我们是否能够使用不重数和公开密钥密码体制（而不是对称密钥密码体制）来解决鉴别问题？这个问题将在本章后面的习题中进行探讨。

8.5 安全电子邮件

在前面的各节中，我们分析了网络安全中的基本问题，包括对称密钥密码体制和公开密钥密码体制、端点鉴别、密钥分发、报文完整性和数字签名。我们现在着手研究如何使