- 教材讨论
  - TJ第7章
  - TC第31章第7、9节

问题1：对称密钥加密和公开密钥加密

- 太公曰：“主与将，有阴符，凡八等。有大胜克敌之符，长一尺。破军擒将之符，长九寸。降城得邑之符，长八寸。却敌报远之符，长七寸。警众坚守之符，长六寸。请粮益兵之符，长五寸。败军亡将之符，长四寸。失利亡士之符，长三寸。诸奉使行符，稽留，若符事闻，泄告者，皆诛之。八符者，主将秘闻，所以阴通言语，不泄中外相知之术。敌虽圣智，莫之能识。”

- 你理解这种加密方法了吗？

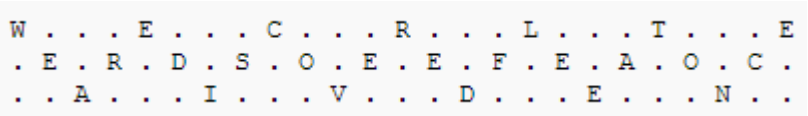
问题1：对称密钥加密和公开密钥加密 (续)

- 斯巴达司令派人给前线送一条这样的腰带：
  KGDEINPKLRIJLFGOKLMNISOJNTVWG

- 你能猜到使用的加密方法吗？
- KGDEINPKLRIJLFGOKLMNISOJNTVWG

问题1：对称密钥加密和公开密钥加密 (续)

- 一条战场快讯：WECRLTEERDSOEEFEAOCAIVDEN
- 你能猜到使用的加密方法吗？



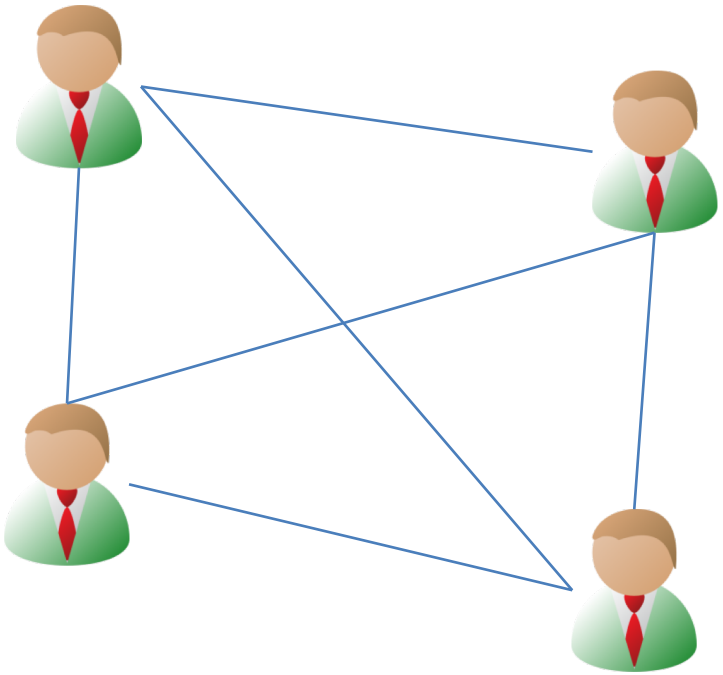
问题1：对称密钥加密和公开密钥加密 (续)

- 对称密钥加密(private/symmetric key cryptography)
- 公开密钥加密(public/asymmetric key cryptography)

- 它们分别是什么含义？
- 各有什么优缺点？
- 你能设计一种新的方式，结合两者的优点吗？
  - Because symmetric key algorithms are nearly always much less computationally intensive than asymmetric ones, it is common to exchange a key using a key-exchange algorithm, then transmit data using that key and a symmetric key algorithm.

问题1：对称密钥加密和公开密钥加密 (续)

- 四个人之间采用对称密钥加密两两间的通讯，你认为需要几个密钥？
- 如果采用公开密钥加密呢？



问题1：对称密钥加密和公开密钥加密 (续)

- 你能简述如何生成RSA的公钥和私钥吗？

1. Select at random two large prime numbers p and q such that $p \neq q$. The primes p and q might be, say, 1024 bits each.
2. Compute $n = pq$.
3. Select a small odd integer e that is relatively prime to $\phi(n)$, which, by equation (31.20), equals $(p-1)(q-1)$.
4. Compute d as the multiplicative inverse of e, modulo $\phi(n)$. (Corollary 31.26 guarantees that d exists and is uniquely defined. We can use the technique of Section 31.4 to compute d, given e and $\phi(n)$.)
5. Publish the pair $P = (e, n)$ as the participant's ***RSA public key***.
6. Keep secret the pair $S = (d, n)$ as the participant's ***RSA secret key***.

- 如何加密、解密？

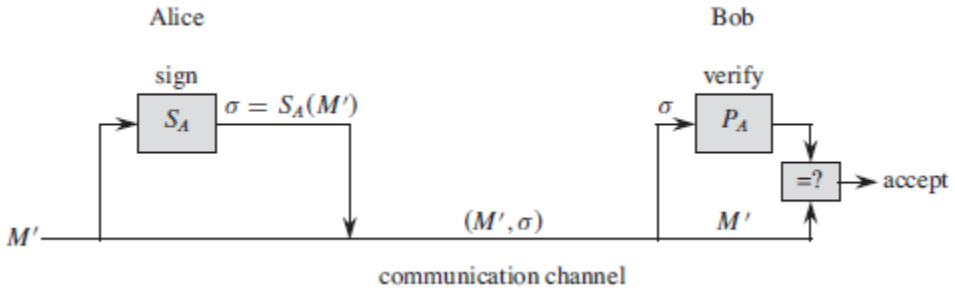
$P(M) = M^e \bmod n$

$S(C) = C^d \bmod n$

- 如果先解密、再加密，会怎么样？
- 破解RSA的关键是什么？为什么？

问题2：数字签名

- 如何利用数字签名分别实现这些目的？（结合实际生活中的签名或盖章）
  - 验证发送者身份
  - 发送者不可抵赖
  - 验证数据完整性
- 如何基于公开密钥加密实现数字签名？和之前的加密/解密过程最大的区别是什么？



- 你有没有办法设计一种基于对称密钥加密的数字签名？

问题2：数字签名 (续)

- 这种身份验证的过程靠谱吗？

S_A(我是A)

A

B

$P_A(S_A$(我是A))=我是A
所以他真是A！

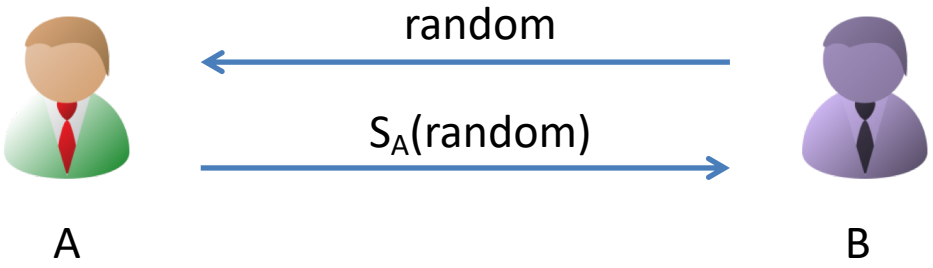
作为一个坏人，你能想出什么办法来冒充A？

从A获取“S_A(我是A)”，向B重放

怎么改进？

问题2：数字签名 (续)

- 改进



$P_A(S_A(random))=random$
所以他真是A！

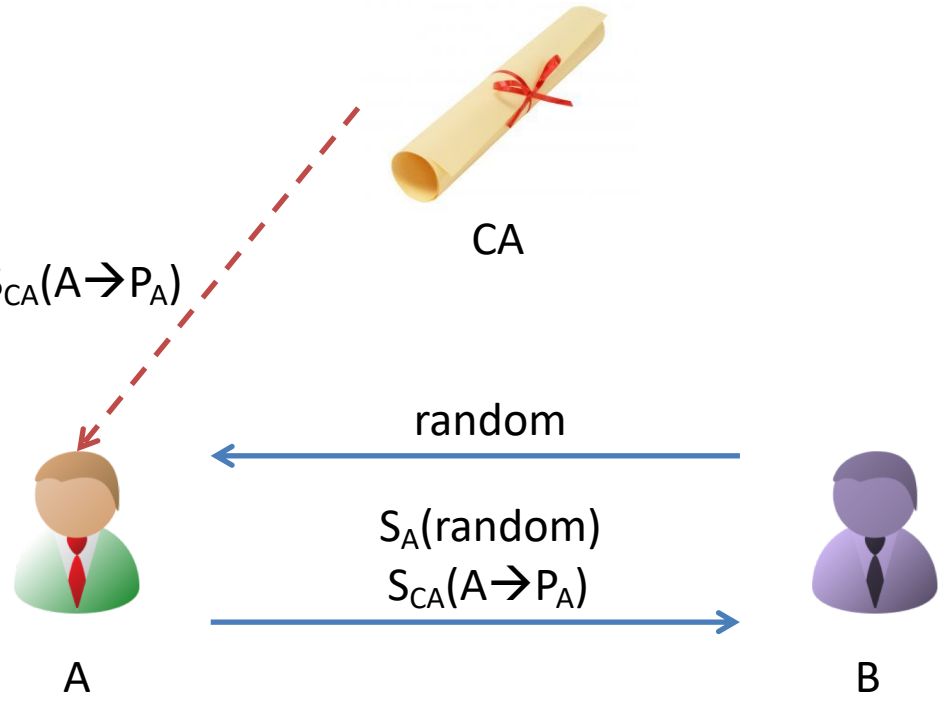
作为一个坏人，你又能想出什么办法来冒充A？

让B具有假冒的P_A

怎么改进？

问题2：数字签名 (续)

- 继续改进



你还能找到漏洞吗？

改进：根证书(+多级认证)

$P_{CA}(S_{CA}(A \rightarrow P_A))=A \rightarrow P_A$
所以A的公钥真是P_A!
$P_A(S_A(random))=random$
所以他真是A!

问题2：数字签名 (续)

- 为了验证数据完整性，除RSA以外，还有更简单的方法吗？
  - 奇偶校验、MD5……
- 与RSA相比，这些方法的优缺点是什么？
  - 数据量小
  - 安全性差
- 如何结合两者的优点？