信息安全导论 hw1

小组成员：

- PB20111704 张宇昂
- PB20111647 鲍润晖
- PB20000103 王炳勋
- PB20111651 何泽昊

Problem 1

选择一种密码系统用于广泛的商业用途，黑客破译密码系统时会造成更大的经济损失，由于境内外可能采用不同的密码系统，可能会造成商业投资的减少，境内外商业交流减少。

美国境外的客户可以通过取得美国政府的证书或者授权，遵守美国法律等来取得密码系统的源代码或使用许可。

Problem 2

块密码的模式的作用是增强密码的安全性。如果每个块都用同样的密钥进行加密，那么相同的块的输出就会一样，所以就会更容易破解。因此，我们需要使用模式来使加密更加复杂且难以破解。

CBC和CTR都是一种格式，可以提高安全性和完整性。

它们的不同在于：

1. CBC使用IV生成输出；而CTR使用nonce和counter生成密钥流
2. CBC是块密码的格式，对每个数据块进行输出；而CTR是流密码的格式，生成的是一串密文流
3. CBC容易收到填充攻击，攻击者可通过特定的padding来获取明文，而CTR不会受到这样的攻击
4. CBC无法并行处理；但CTR可以

Problem 3

冲突的必然发生是因为哈希函数将任意长输入映射为定长输出，前者是无限的而后者是有限的，所以必然会有两个不同的输入被映射到同样的输出
哈希函数的抗冲突(collision resistance)可以使冲突不构成安全问题。攻击者即使知道了原有明文和它对应的哈希值，也很难构造出一个新的明文，使它的哈希值和之前一样。因为哈希函数应该是不可逆的，所以攻击者只能遍历所有输入的明文，来获得它们的哈希值，并与目标哈希值对比是否相等，这样才能找到一个冲突的新明文。而它的代价是很大的

Problem 4

https://ustc.edu.cn/：

通过浏览器中的证书查看器，可查询证书的相关信息

根CA：USERTrust RSA Certification Authority

中间CA：ZeroSSL RSA Domain Secure Site CA

证书到期时间：2023/6/15 GMT+8 07:59:59

导出证书到.pem文件中，控制台中输入指令：openssl x509 -noout -modulus -in ustc.edu.cn.pem | openssl md5

得到md5 hash值为：6840b8c2652f9c0d65e7ebc518f6e915

控制台中输入指令：openssl x509 -noout -modulus -in ustc.edu.cn.pem | openssl sha256

得到sha256 hash值为：
50237fb8f170d610167c53f0ac1f339574a144d549dd41688d17886ff2944437

```
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in ustc.edu.cn
.pem | openssl md5
(stdin)= 6840b8c2652f9c0d65e7ebc518f6e915
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in ustc.edu.cn
.pem | openssl sha256
(stdin)= 50237fb8f170d610167c53f0ac1f339574a144d549dd41688d17886ff29
44437
```

www.12306.cn:

根CA：CFCA EV ROOT

中间CA：CFCA OV OCA

证书到期时间：2023/10/24 GMT+8 09:49:31

导出证书到.pem文件中，控制台中输入指令：openssl x509 -noout -modulus -in www.12306.cn.pem | openssl md5

得到md5 hash值为：754abefbed19f2df687f0cbe70aab41e

控制台中输入指令：openssl x509 -noout -modulus -in www.12306.cn.pem | openssl sha256

得到sha256 hash值为：
c5f4cf3bef04e2799b8a81e224e5d89a82ca43b9b9a866c1e9871e349243e8bc

```
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in www.12306.c
n.pem | openssl md5
(stdin)= 754abefbed19f2df687f0cbe70aab41e
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in www.12306.c
n.pem | openssl sha256
(stdin)= c5f4cf3bef04e2799b8a81e224e5d89a82ca43b9b9a866c1e9871e34924
3e8bc
```

www.bing.com

根CA：Baltimore CyberTrust Root

中间CA：Microsoft RSA TLS CA 02

证书到期时间：2023/8/16 GMT+8 11:47:45

导出证书到.pem文件中，控制台中输入指令：openssl x509 -noout -modulus -in www.bing.com.pem | openssl md5

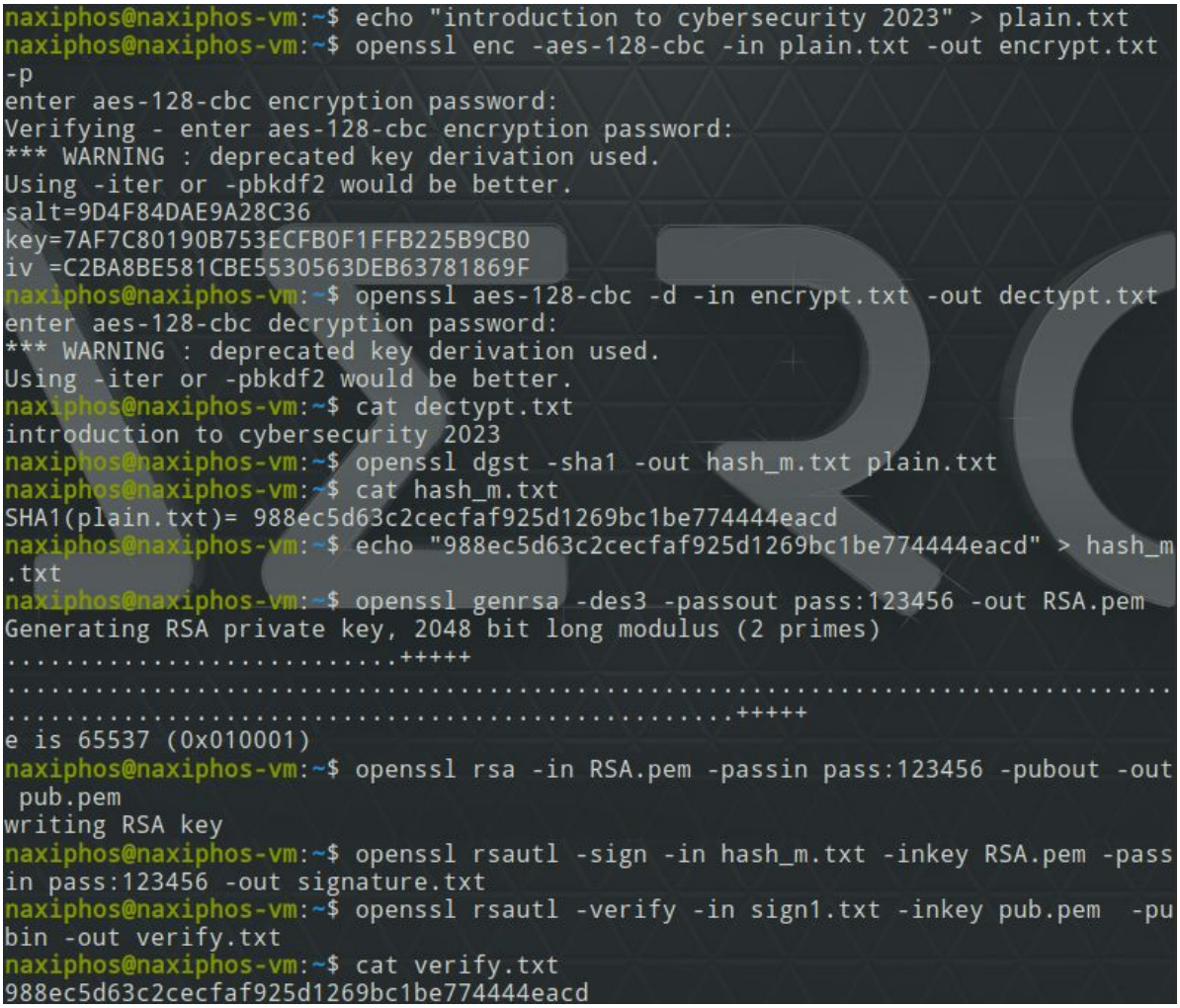
得到md5 hash值为：e1f0137879d09e462f3a7ea62ff7b6b1

控制台中输入指令：openssl x509 -noout -modulus -in www.bing.com.pem | openssl sha256

得到sha256 hash值为：
428bd67704d7181d158c383ed0c8529a9d1f5c4e2935a007ac1a891f9c382658

```
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in www.bing.co
m.pem | openssl md5
(stdin)= e1f0137879d09e462f3a7ea62ff7b6b1
wbxun@ubuntu:~/Desktop$ openssl x509 -noout -modulus -in www.bing.co
m.pem | openssl sha256
(stdin)= 428bd67704d7181d158c383ed0c8529a9d1f5c4e2935a007ac1a891f9c3
82658
```

Problem 5



Problem 6

Summary

密码学的主要问题为加密和认证两方面，这篇论文也是从加密和认证两方面来介绍并探讨传统的加密算法以及新提出的非对称加密算法

我们主要关注Section II~Section VI提及加密算法的部分

Section II 传统密码学

传统密码学中，两大问题为：

- 隐私系统：防止在不安全的信道上传输消息时被未经授权者从中获取信息，确保发送者发送的消息只能被预期的接收者读取。
- 身份验证：防止未经授权将消息注入公共通道，从而确保消息的接收者验证其发送者的合法性。包括消息身份验证和用户身份验证

传统密码系统的流程为：

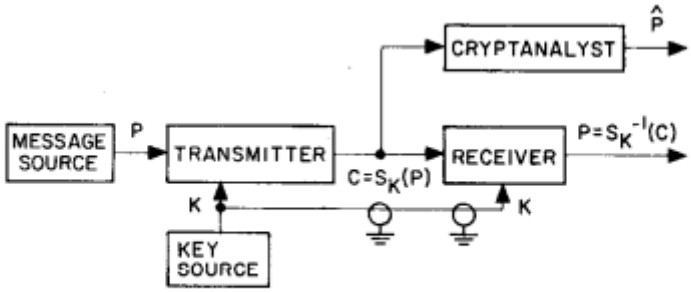


Fig. 1. Flow of information in conventional cryptographic system.

- 发送者生成消息P
- 通过可逆变化S_k加密消息P得到密文：$C = S_k(P)$
- 通过不安全信道传输密文
- 接收者通过仅双方已知的密钥进行解密$S_k^{-1}(P)$得到明文

Section III 公钥加密

公钥密码的可逆变换的算法为：

- 加密：$E_k : \{M\} \to \{M\}$
- 解密：$E = D_k : \{M\} \to \{M\}$

M表示明文/密文的所在空间（相同的）

公钥密码系统的流程为：

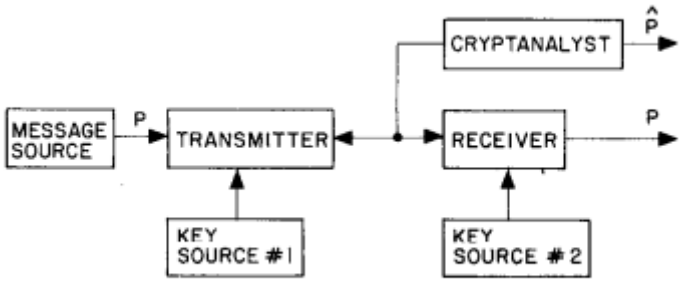


Fig. 2. Flow of information in public key system.

- 用户生成一对公钥-私钥对(E-D)
- 将加密密钥与用户信息公开
- 其他的任何用户可以通过获得公共目录中的公钥与用户地址，使用公钥加密并按其地址发送给目标用户
- 私钥拥有者收到信息并解密

该算法的安全性建立在密钥分发技术在**有限域上离散对数的计算的单向性**

Section IV 单向认证

单向认证中最重要的是找到一个单向函数$f(x)$，其满足对$f(x)$的计算非常容易，但是在**计算意义上**计算$f^{-1}(x)$是不可行的

文章中介绍了三种单向函数的设计来进行单向认证

1. 通过单向函数计算的单向性：
    - A使用不公开的私钥进行加密，得到密文发送给B。
    - B接收到了A发送的密文，使用A公开的公钥进行解密，得到对应的信息。
    - 由于私钥的不公开性，因此只有A自身发送信息才具有这样的性质。

2. (Leslie Lamport)通过K维二进制向量的单向函数到K维空间的映射
    - 产生2N bit的随机二进制向量：$x_1, X_1, \ldots, x_N, X_N$
    - 分别计算x, X的单向函数：$y_1, Y_1, \ldots, y_N, Y_N$
    - 将得到的y, Y发送，将x, X保密
    - 发送者逐个bit发送，若为0发送x，否则发送X
    - 接收者接收后根据是y/Y判断是0/1
3. 通过发送时间构造一个单向函数$f^T(x)$
    - 当前的时间点为t，对应的身份验证器为$f^{T-t}(X)$
    - 用户计算对应的$f^t(X)$提交到系统
    - 系统通过计算得到 $f^T(X)$进行验证

Section V 单向陷门函数

问题的相关性：**已知明文攻击安全的密码算法和单向函数问题之间是相关的**

假设有一个能够抵抗已知明文攻击的系统S_k，已知明文为一个固定常量P，于是从密钥得到密文的映射就可以作为一个单向函数f

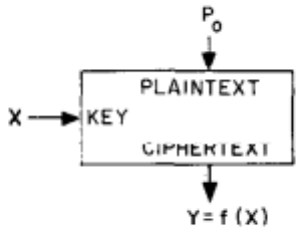


Fig. 3. Secure cryptosystem used as one-way function.

但是这种方法破坏了已知明文攻击下的安全性，并且增加了计算的复杂度。

于是作者提到可以用公钥密码系统生成单向认证系统，这是因为公钥系统实际上是一个单向的陷门函数，但是其不是真实意义上的单向函数，因为存在对应的逆运算。但是通过正向的函数去寻找一个对应的逆运算的算法在计算上是不可行的，只有通过特定的陷门信息(密钥)才能得到。得到对应的单向函数后就可以构建对应的单向认证系统。

最后介绍了单项陷门密码算法来产生公钥分配的算法：

- 用户1任选一个密钥，用用户2公布的包含陷门信息的密钥加密
- 用户1给用户2发送密文
- 用户2解密得到密钥
- 建立公钥系统

Section VI 计算复杂度

在这一段中作者提到最关键的一点是：一个加密/解密算法若可以在多项式时间内完成，那么密码分析的难度不会大于NP；同时作者也指出现代密码算法的安全性是基于**计算意义上的不可行**定义的

Reviews

这篇论文总结了在当时的加密算法，但是同时提到所有的传统加密算法已被破解，以及现有的可行的加密算法都需要一个可信的公共信道去传递一个双方可知的公钥，然而这样并不方便。

所以作者提出了一个一种新的算法，在这个算法中，一个用于加密的公钥并非双方私有的，而是可以被公开、分享的；但是同时也需要一个永远不需要离开接收装置的私钥进行解密。这个算法的重要意义在于，它表明**非对称加密/公共密钥加密是可行的**

正如题目所言，作者在这篇论文中为密码学提供了一种新的发展方向：利用计算复杂性问题可以构造陷门单向函数，进一步可以构造公钥密码学，而公钥密码学可以实现加密和认证功能。后续的研究一步步地实现了这些功能。

Advantage

1. 解决了密钥管理问题，当用户数量级很大时密钥也不会如同公钥加密算法一般向外扩散
2. 创新性的表明了非对称加密的可行性，提高了传输的安全性，很难进行破解

Disadvantage

加密解密都需要一定的计算量，速度较慢，算法复杂度较高