实验总结

现对本学期密码学学过实验中的五个实验来谈谈自己的感受和总结。

1. 古典密码。

首先对我们的第一堂实验课进行总结，这是一次打开我学习密码学的大门。

1. 基本介绍：

古典密码-移位密码。算法是不改变明文字符，而是按照某一规则重新排列消息中的比特或字符顺序，从而是吸纳明文信息的加密。将明文中的字母按照给定的顺序安排在一个矩阵中，然后用根据秘钥提供的顺序重新组合矩阵中的字母，从而形成密文。其解密过程是根据秘钥的字母数作为列数，将密文按照列，行的顺序写出，再根据秘钥给出的矩阵置换产生新的矩阵，从而得到明文。**它的加解密过程可以用以下方式表示：**

**C=Ek(s)=(s+k) mod n,**

**S=Dk(c)=(c-k) mod n,**

**其中，c表示密文字符，s表示明文字符，k表示移位的数字，n表示代换字符集的字符总个数，当字符集为26个字母时的移位算法就是凯撒密码。**

1. **实践体会移位密码--凯撒密码：**

**设置一个字符串：“i love you” 设置key 为3。加密的密文为：**l oryh brx。这个就是将每个字符向后移位3个位置，对照26个字母表即可得出密文。

1. 感受。

第一次接触密码学，通过这个实验感觉到密码学的魅力和有趣。

二 DES算法加密解密过程

DES是对称的，它使用同一个密钥来加密和解密数据。

所需数据

key：8个字节共64位的工作密钥

data：8个字节共64位的需要被加密或被解密的数据

加密过程

明文64位->初始置换IP->16轮加密变换->逆初始置换IP-1->密文。

解密过程：

DES的解密过程和DES的加密过程基本类似，唯一的不同是将16轮的子密钥序列的顺序反过来，即在加密过程中所用的加密密钥k1，...，k16在解密过程中需要转换成k16，...，k1。

DES算法整理：

分组加密算法：以64bit为分组。输入64bit明文，输出64bit密文。

对称算法：加密和解密使用同一秘钥。

有效秘钥长度：56bit秘钥，通常表示为64位数，但每个第8位用作奇偶校验，可以忽略。

代替和置换：DES算法是两种加密技术的组合：混乱和扩散。

易于实现：DES算法只是使用了标准的算术和逻辑运算，其作用的数最多也只有64 位

DES算法的特点：DES算法具有极高安全性，除了用[穷举搜索法](https://baike.baidu.com/item/%E7%A9%B7%E4%B8%BE%E6%90%9C%E7%B4%A2%E6%B3%95" \t "https://baike.baidu.com/item/des%E7%AE%97%E6%B3%95/_blank)对DES算法进行攻击外，还没有发现更有效的办法。而56位长的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "https://baike.baidu.com/item/des%E7%AE%97%E6%B3%95/_blank)的穷举空间为2^56，这意味着如果一台计算机的速度是每一秒钟检测一百万个密钥，则它搜索完全部密钥就需要将近2285年的时间，可见，这是难以实现的。所以DES算法被攻破的成本很大，安全性较高。

感受：在理论课学习的过程中刚开始还是很难理解加密过程的子密钥的求法，对于ip置换也一知半解，但是通过本次实验，对DES的了解更加透彻了，也帮助了对理论课程中DES的学习。

1. 端口扫描与明文嗅探

本次实验我们使用Linux系统作为扫描机，windows作为靶子。

1. 端口扫描

通过端口扫描，可以得到许多有用的信息，从而发现系统的安全漏洞。端口扫描技术的原理是端口扫描向目标主机的 TCP/IP服务端口发送探测数据包，并记录目标主机的响应。通过分析响应来判断服务端口是打开还是关闭，就可以得知端口提供的服务或信息。

我们此次使用的工具有：**nmap。**

**实践体会：**

**打开主机Linux：**nmap –sS windows-ip 进行端口扫描。查看当前windows端口开放情况。（此命令是TCP SYN scanning扫描方式）该方式发送SYN到目标端口，如果收到SYN/ACK回复，那么判断端口是开放的；如果收到RST包，说明该端口是关闭的。如果没有收到回复，那么判断该端口被屏蔽。

输入nmap –sT -A windows-ip（-sT表示 TCP connect scanning，-A表示显示操作系统及版本探测）命令。TCP connect方式使用系统网络API connect向目标主机的端口发起连接，如果无法连接，说明该端口关闭。该方式扫描速度比较慢，而且由于建立完整的TCP连接会在目标机上留下记录信息，不够隐蔽。

总结：

我们可以用nmap工具进行多个方式进行扫描，以到达各种情况下的扫描。这个实验也是与我们专业非常相关，本次实验中，我觉得做起来较为简单，有趣。

1. 明文嗅探

如果此时要实现在指定嗅探器上实现整个网络的嗅探目的，则要在交换机上启用端口镜像功能，这种功能允许你将交换机中的一个端口设置为端口镜像模式，然后再指定要被镜像的交换机端口关联到这个指定了镜像功能的端口上。完成设置后，这些被镜像的交换机端口中的流量将会同时复制一份到镜像端口上。

开启linux靶机，在终端使用工具：sniffit进行嗅探。然后对此我们通过分析这些流量找到我们想要找到的数据。

总结：

通过这个实验，我感觉到信息安全的重要性。

1. Rsa算法

RSA公开密钥密码体制的原理是：根据数论，寻求两个大素数比较简单，而将它们的乘积进行因式分解却极其困难，因此可以将乘积公开作为加密密钥。

特点：

运算速度：

由于进行的都是大数计算，使得RSA最快的情况也比DES慢上好几倍，无论是软件还是硬件实现。速度一直是RSA的缺陷。一般来说只用于少量数据加密。RSA的速度比对应同样安全级别的对称密码算法要慢1000倍左右 。

算法攻击

迄今为止，对RSA的攻击已经很多，但都没有对它构成真正的威胁。在这里，我们讨论一些典型的攻击方法 。

感受：通过这次的实验，对rsa的理论和实际操作有了更多的了解了。

1. 鸽子实验

服务端和客户端。植入对方电脑的是服务端，而黑客正是利用客户端进入运行了服务端的电脑。运行了木马程序的服务端以后，会产生一个有着容易迷惑用户的名称的进程，暗中打开端口，向指定地点发送数据（如网络游戏的密码，实时通信软件密码和用户上网密码等），黑客甚至可以利用这些打开的端口进入电脑系统。这样的木马对于我们是一种严重的危害。

实验步骤：

将灰鸽子、文件捆绑器、飞鸽在windows下D盘->攻防工具包->灰鸽子木马。

同学a开启windows、linux靶机并连接，windows靶机上生成木马并捆绑，通过桌面的SSH Secure File Transfer Client，把木马上传到linux的/opt/lampp/htdocs/hgz/下；

同学b开启windows靶机并连接，通过地址访问同学a的linux上的服务器，下载木马并运行；

控制放：通过使用灰鸽子对中了木马的同学b的windows靶机进行控制。

感受：此次实验让我看到了木马的恐怖之处，对于我们的另一边而言，我们是没有秘密可言。

最后的感受：通过本学期的实验，自己对密码学有了更多的了解和并对密码学产生了更多的兴趣，有些实验有难度，但是通过老师和同学们的帮助下，还是圆满完成了每次的实验，收获颇丰。