

“网络安全综合实验（I）”实验报告

**题目： 加密与解密实验**

院 系 网络空间安全学院

专业班级 xxxx

姓 名 xxx

学 号 xxxx

日 期 2021 年 6 月

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 评分表 | 加密与解密实验报告评分 （50分） | | | | 完成（50分） | 成绩 | |
| 评分 项目 | 过程 | 分析与小结 | 撰写 | 创新 | 完成任务 | 合计 | 教师 |
| 分值 | 20 | 20 | 5 | 5 | 50 | 100 | 签名 |
| 评分 |  |  |  |  |  |  |  |

报告要求及评分规则

1.请参考模板给出的格式，包括行距、字体、段落格式等，完成报告；

2.报告封面保持一致(如实填写基本信息及完成日期)；

3.实验报告内容应包括：封面、评分规则、实验过程记录、实验问题分析与总结、参考文献及资料列表；

4提交：电子材料应包括本实验电子版（doc）、实验参考文献资料的电子资源文件（pdf）。

5.主要考察能力： 实验动手能力、问题分析与归纳能力、文档规范撰写能力、创新能力；

6.其他要求：可按指导老师要求的时间和提交方式提交；每次课实验报告可以单独提交；如果需要最终纸质报告的，可去掉模板中说明文字（斜体）之后，双面打印；

7.总评分=课程每次实验分数之和/课程实验次数。

其中：

每次实验分=实验完成分（50分）+实验报告分（50分）-扣分+加分；

计算方法：

实验完成分=（完成任务数量/总任务数量）\*50；

实验报告分=以下1-4项合计；

1）过程（要求：实验过程完整、清晰）(满分20)（优秀：18+ 良好：16+ 一般：14+）；

2）问题分析与小结（要求：有条理、细致）（满分20）（优秀：18+ 良好：16+ 一般14+）；

3）撰写（要求：语句通畅、格式规范）（满分5）（优秀：4+ 良好：3+ 一般：2+）；

4）创新（要求：见解独到、有创意）（满分5）（优秀：4+ 良好：3+ 一般：2+）；

扣分=报告迟交天数\*2分（满分10分）  （发现雷同抄袭的内容，该次实验不得分）；

加分=搜集整理与实验相关的学习资料作为附件，资料能帮助同学更好掌握相关知识的；获得同组人互评优秀的；提交最终报告时间为班级前3名。（满分5分）（项数\*1分）；

同组评价：

本实验中你的同组人姓名： xxx 学号： xxxxx

你给同组人评价： 优 （优 良 中 及格）；

请描述具体理由： 能比较透彻地理解实验指导书内容，动手操作能力强，遇到问题、困难，敢于并积极解决。善于利用搜索引擎来寻找问题的解决办法，能独立解决命令执行中的大多数问

题。能较好掌握本次实验的内容和原理，并加以实践应用。

# 

# 加密与解密实验（通关）

## 实验环境及要求

### 实验平台及说明

虚拟机：Vmware 15或者VirtualBox；

操作系统：虚拟机内安装kali Linux；

本实验2人一组，同组成员： xxx 、 xxxx ；

### 实验场景设置

还记得上次数据库关卡吗？你撤销了Bob对Galaxy数据库的访问权限，并对数据进行了保护，包括用MD5、AES等。这次Bob通过一些渗透工具的使用，发现了一些不太安全的因素，会提醒你注意；另外，假如你被派往上海出差，你和Bob远程通信时，采取适当的安全防护，比如文件加密、邮件附件加密等。

## 过程记录/实验任务（共10个关卡，25个小任务）

* + 1. **关卡1 达芬奇密码**

**1）密码恢复**

公司内部用于培训数据的资料，被压缩软件winrar压缩，并加上了密码进行保护，但这个文件是已经辞职的管理员留下的，原密码已经丢失；请你想办法恢复该密码。

提示：采用工具Advanced Archive Password Recovery（ARCHPR）进行密码恢复：

*提示信息：如果直接恢复15分钟，仍未成功，可以利用信息“Call the Hospital of HUST”削弱破解难度。*

**解答：**

（a）采用工具 Advanced Archive Password Recovery（ARCHPR）进行密码恢复，并利用信息“密码为8位数字，以 87开头，0结尾”削弱破解难度，进行暴力破解，如图1-1所示。

（b）成功恢复密码，如图1-2所示。

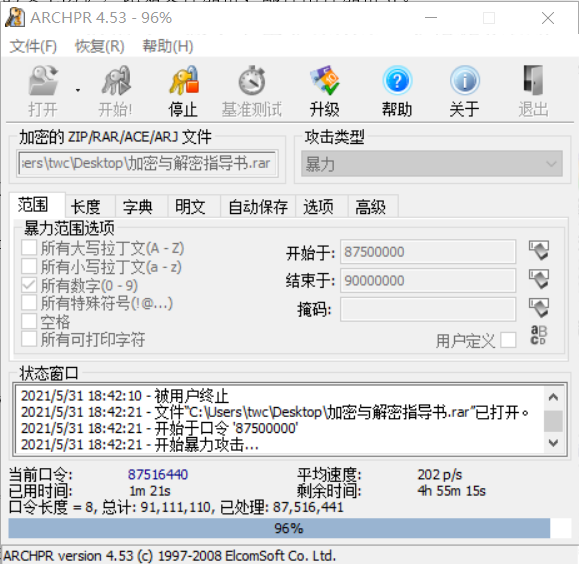


图1-1 缩小范围恢复密码的截图

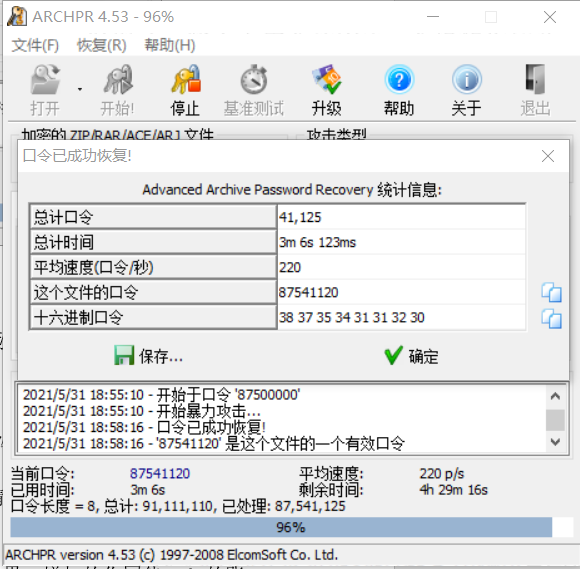


图1-2 成功恢复密码的截图

* + 1. **关卡2 友谊的小船**

**2）互换文件破解**

同组两位同学，各自压缩并设置密码保护某文件，相互交换后，尝试破解；

**解答：**

接收同伴的压缩文件，开始破解，如图1-3所示。

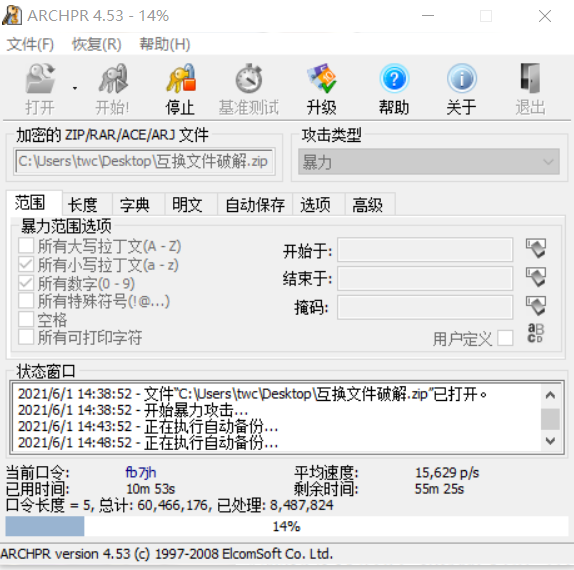


图1-3 互换文件破解的截图

**3）强度削弱**

如果10分钟后，某一方未能完成关卡2，可请对方给出提示以缩小尝试的范围；请记录对方给出的提示。

**解答：**

10分钟后未成功破解，同伴给出提示：7位小写字母，zx开头，缩小范围，成功破解，如图1-4所示。

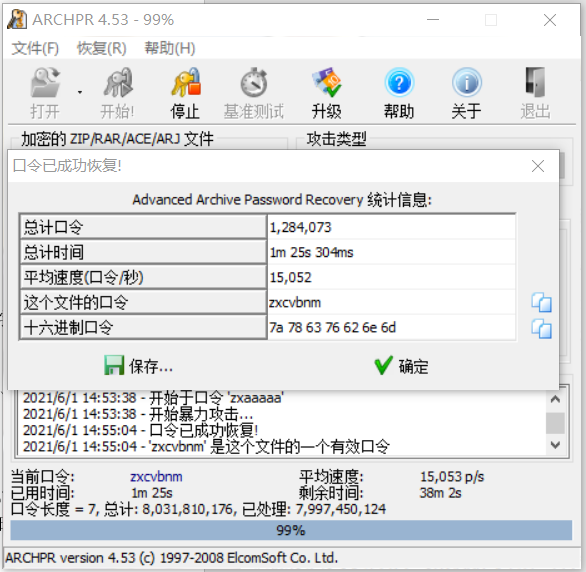


图1-4 提示后再次破解的截图

* + 1. **关卡3 好记性不如烂笔头**

口令恢复：前面Linux基础关卡中，kali系统里，增加的你同伴Bob的账号吗？ 请Bob自己登录系统后，修改自己的密码；现在，假设Bob太长时间没有登录系统，遗忘了该密码。

如果希望找回原始密码，Kali系统有这项功能吗？

答案： 没有，但可以重置系统密码，详细参考附件 ；

请你按小任务4）-7）找回你同组人Bob的原始密码。

**4）获取密文**

从虚拟机中/etc/shadow中获得该同学用户密码的密文；

**解答：**

$6$的含义是使用SHA-512算法对用户密码进行加密。

（a）执行命令sudo tail -n 5 /etc/shadow | grep

wangzhuo > ./wangzhuo.pass将用户密码的密文写入wangzhuo.pass文件中，如图3-1所示。



图3-1 获得用户密码密文的截图

（b）执行命令cat ./wangzhuo.pass查看密码密文，如图3-2所示。

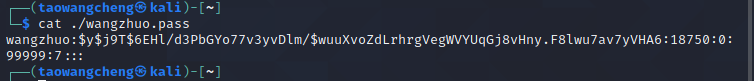


图3-2 查看密文的截图

（c）执行命令vim wangzhuo.pass，删除无关内容，如图3-3所示。

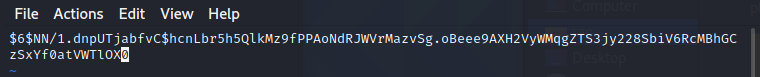


图3-3 修改密文的截图

**5）暴力破解**

使用kali中的Hashcat工具，用掩码模式，尝试暴力破解；

如果破解10分钟，还没有结果，

记录保存当时的状态截图，记录如下；

之后，再尝试用用字典破解。

**解答：**

（a）执行命令hashcat -m 1800 wangzhuo.pass -a 3 ?l?l?d?d?d?d?d?d –force进行暴力破解，如图3-4所示。

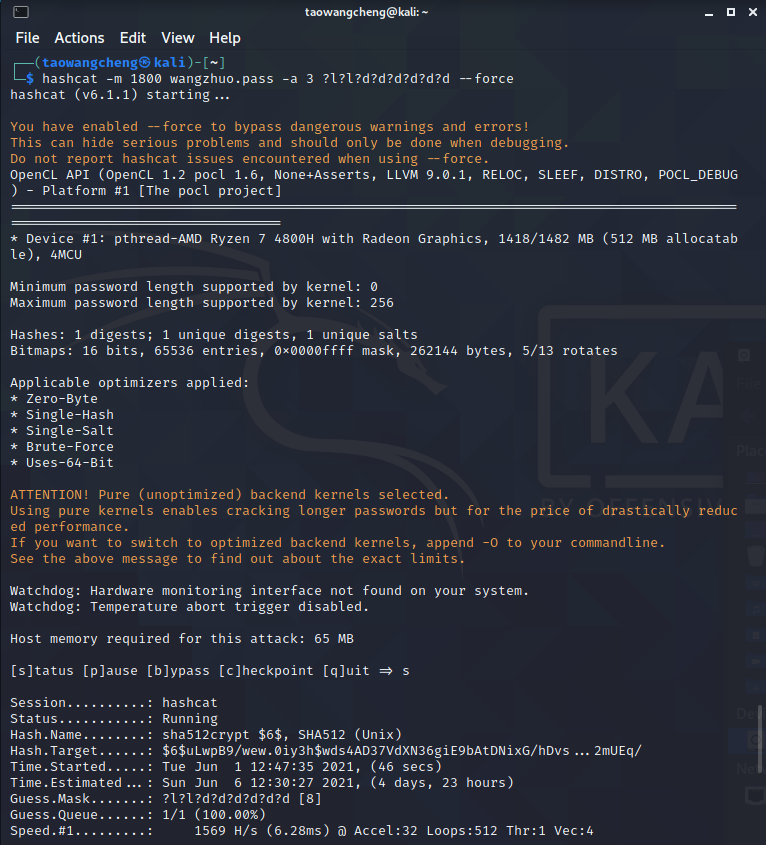


图3-4 暴力破解的截图

（b）运行一段时间后未破解成功，如图3-5所示。

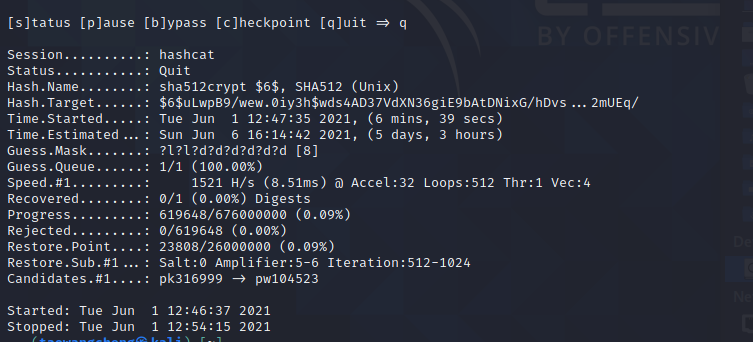


图3-5 暴力破解时间过长的截图

**6）生成字典**

请同组成员，给出提示信息，记录提示信息：

密码以bo开头。

根据提示信息，使用字典的工具：crunch，生成字典。

**解答：**

执行命令crunch 8 8 -t bo%%%%%% -o wordlist.txt生成字典，如图3-6所示。

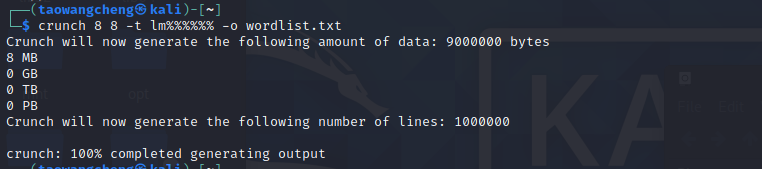


图3-6 生成字典的截图

**7）用字典继续完成破解**

如果5分钟不能破解成功，先保存截图，然后请对方给出更多提示信息，再尝试。

**解答：**

（a）执行命令hashcat -m 1800 -a 0 -o found.txt wangzhuo.pass wordlist.txt –force

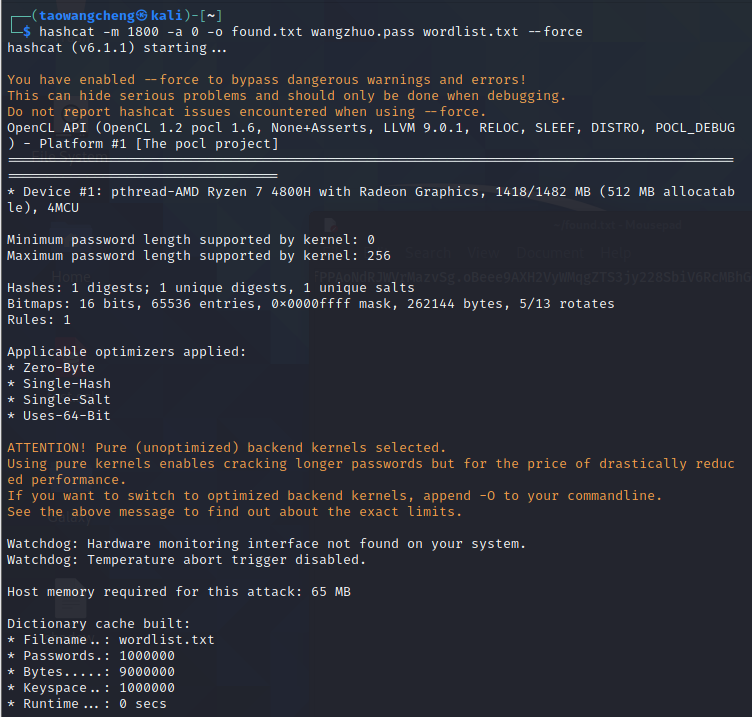


图3-7 字典破解的截图

（b）字典破解成功，如图3-8所示，原始密码在found.txt文件中，如图3-9所示。

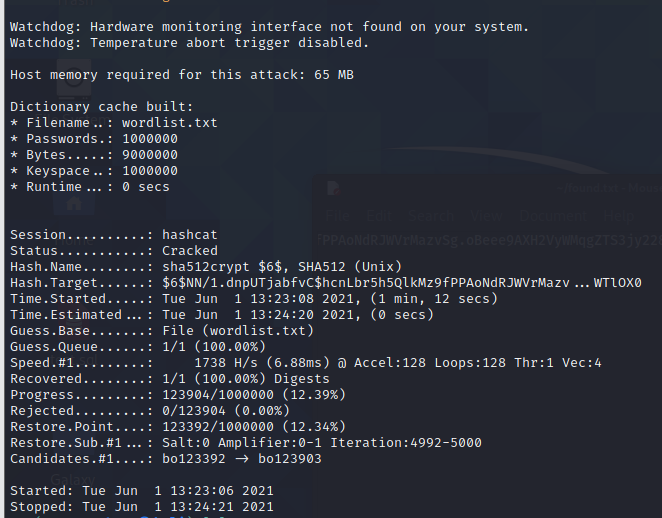


图3-8 字典破解成功的截图

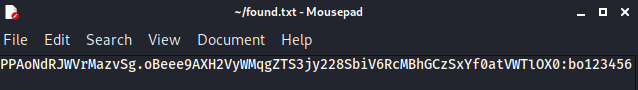


图3-9 原始密码的截图

**8）独立思考**

回答问题：hashcat能否破解数据库安全基础关卡中，采用MD5、AES加密了某些字段？

填写你的答案： hashcat能够破解采用MD5加密的字段，不能破解采用AES加密的字段。

你同组人同意吗？ 同意。

尝试用前面的工具，进行实验，证明你的想法，并记录： 破解MD5字段的过程如下：

**解答：**

（a）执行命令hashcat -a 3 e10adc3949ba59abbe56e057f20f883e

?d?d?d?d?d?d –force破解MD5字段，如图3-10所示。

（b）成功破解，显示为123456，如图3-11所示。

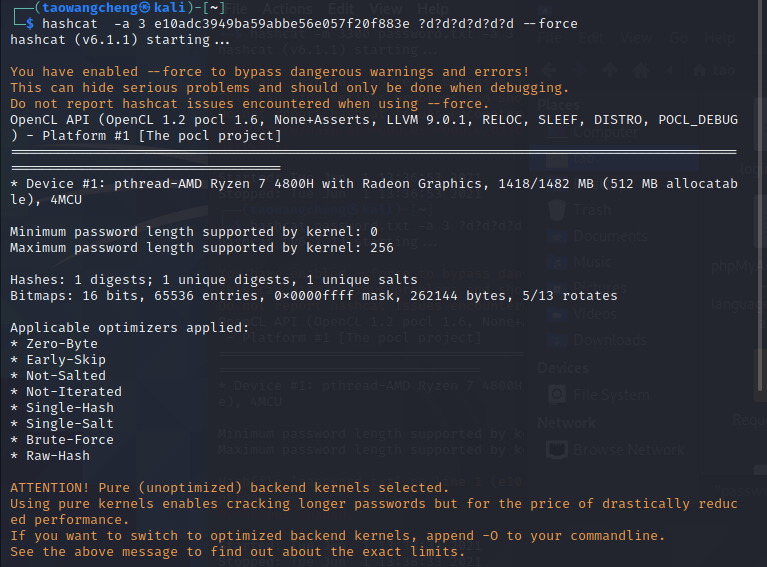


图3-10 MD5字段破解的截图

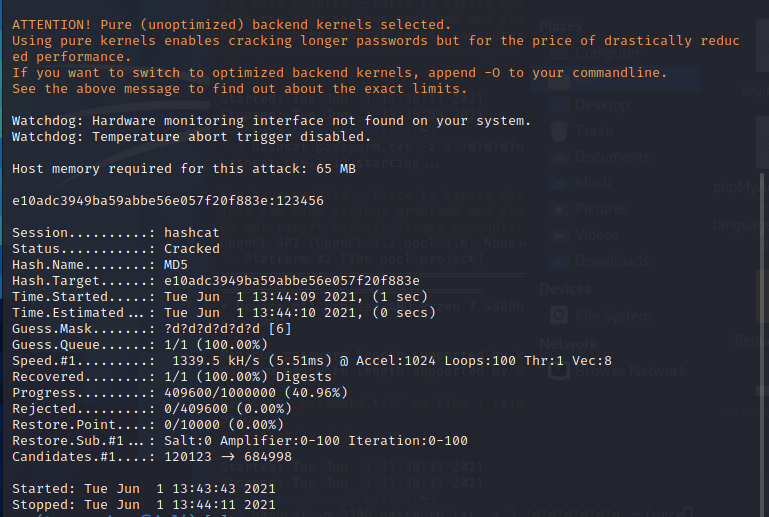


图3-11 MD5字段成功破解的截图

* + 1. **关卡4 十八般兵刃**

天下安全，唯密不破；

从前面的过程中，你应该发现了安全中的一些问题或者原则。现在需要用更强大的工具，对一些内容进行安全防护。掌握好openssl及密码算法的名称。

openssl 使用

**9）查看已安装的openssl版本**

通过命令查看并记录版本信息。

**解答：**

执行命令openssl version查看简略版本信息，执行命令openssl version -a查看完整版本信息，如图4-1所示。

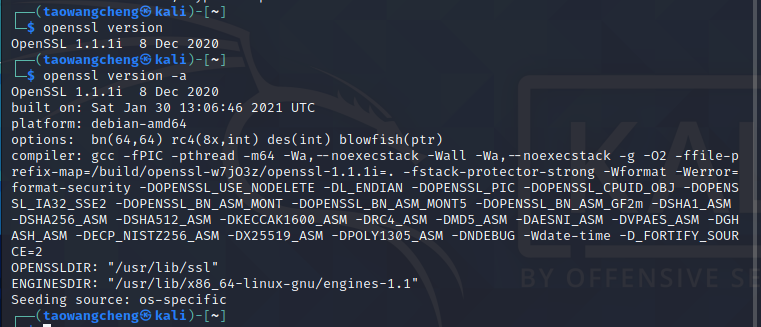


图4-1 查看openssl版本信息的截图

**10）查看帮助**

查看openssl联机帮助，了解其功能。

**解答：**

执行命令openssl help查看帮助，如图4-2所示。

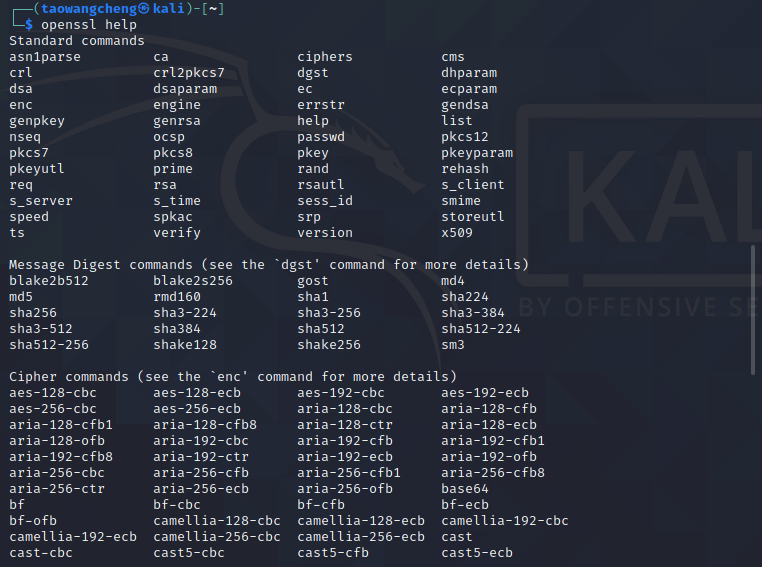


图4-2 查看openssl联机帮助的截图

查看帮助后，回答问题：openssl命令分为哪几类命令？

1、Standard commands ；

2、Message Digest commands ；

3、Cipher commands ；

**11）加密速度测试** Benchmarking

对比一下你机器使用openssl加密的速度与你同伴的差异，记录两台主机的配置，虚拟机的配置，同一种算法的速度：

**解答：**

（a）两台主机的配置如图4-3、4-4所示。



图4-3 自己主机配置的截图

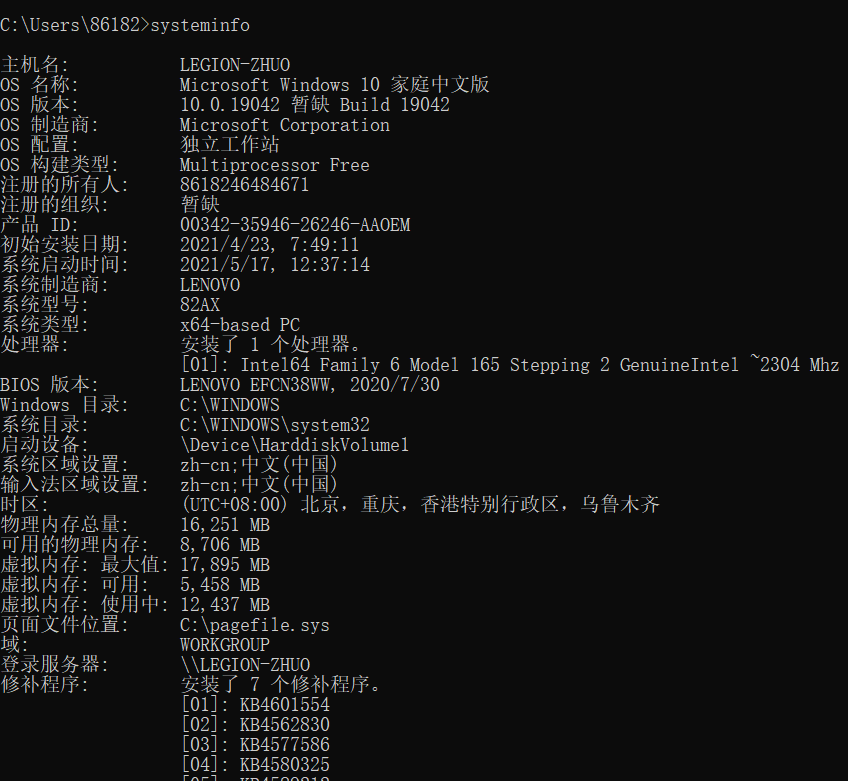


图4-4 同伴主机配置的截图

（b）两台虚拟机的配置如图4-5、4-6所示。

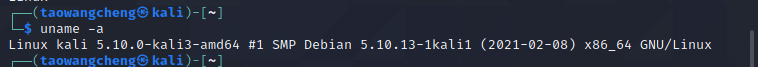


图4-5 自己虚拟机配置的截图

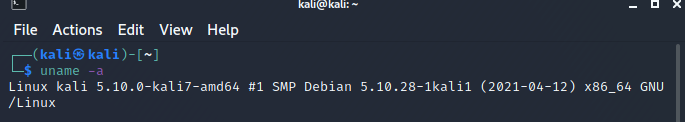


图4-6 同伴虚拟机配置的截图

（c）在两台机器上执行命令openssl speed md5记录速度，如图4-7、4-8所示。

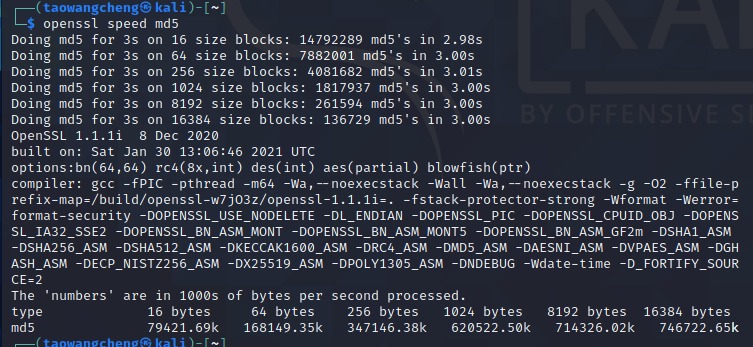


图4-7 自己机器速度测试的截图

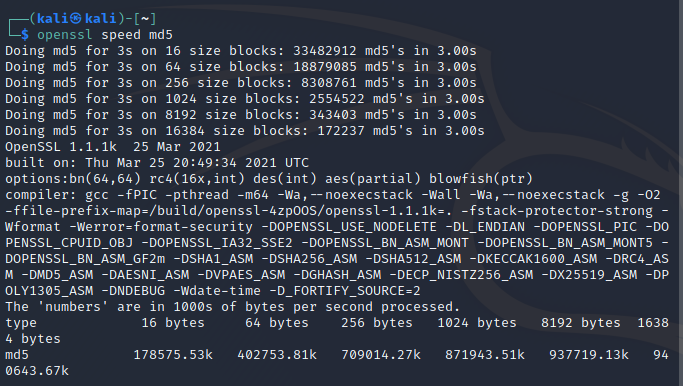


图4-8 同伴机器速度测试的截图

* + 1. **关卡5 Digest摘要算法**

**12）被篡改了吗？**

从课程群下载的phpmyadmin压缩包，你能确定是官方发布的正式版吗?有没有被篡改过？想办法利用openssl来检测一下，说明你测试的过程和结果。

**解答：**

（a）从phpMyAdmin官网上下载SHA256校验码，如图5-1所示。

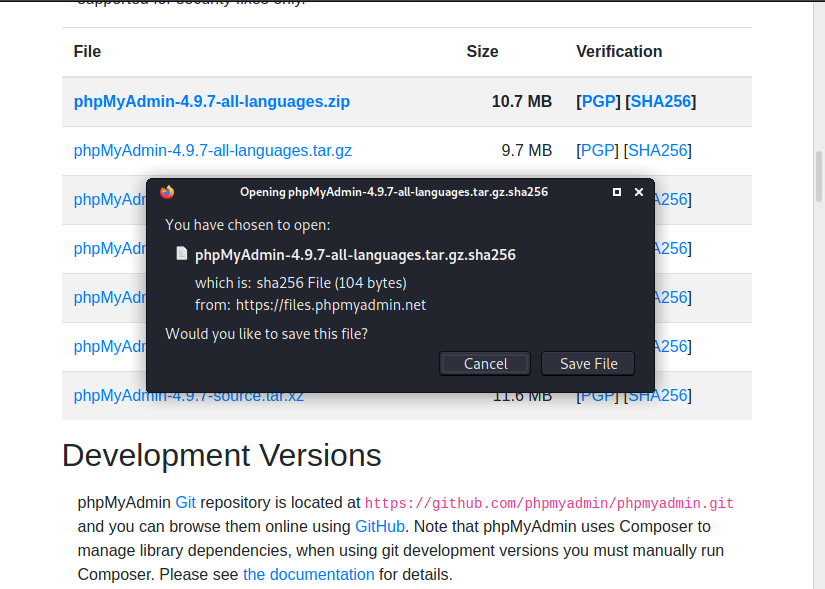


图5-1 官网下载校验码的截图

（b）使用openssl，选择sha256算法，生成校验码，如图5-2所示。

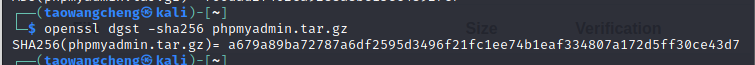


图5-2 openssl生成校验码的截图

（c）比较校验码，发现一致，如图5-3所示。

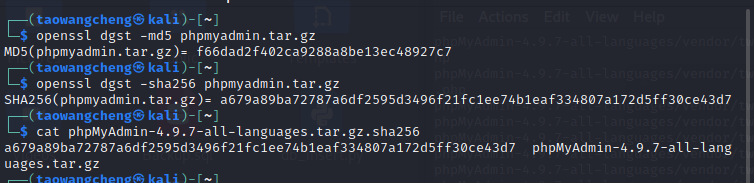


图5-3 比较校验码的截图

**13） Shell编程**

灵活运用你掌握的技术，假设需要你写一个小程序，读取用户输入的一些信息，并加密它们。比如写一个模拟登录过程的脚本：

用vi或其他编辑器写个shell脚本，功能：读取用户键盘输入的姓名、密码、认证串三个符号串参数，然后用openssl命令，计算输入的三项内容拼接成的符号串对应的md5值，输出到控制台屏幕上。

运行你的脚本文件，输入你自己的姓名等信息，创建摘要并记录结果。

运行并测试你写的脚本，总结你的发现。

**解答：**

（a）用vim编写shell脚本，实现相关功能，如图5-4所示。

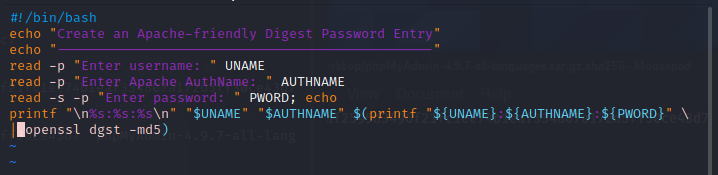


图5-4 编写脚本的截图

（b）执行命令chmod +x login\_md5将脚本文件权限修改为可执行，如图5-5所示。



图5-5 修改脚本文件权限的截图

（c）执行命令./login\_md5运行脚本，如图5-6所示。

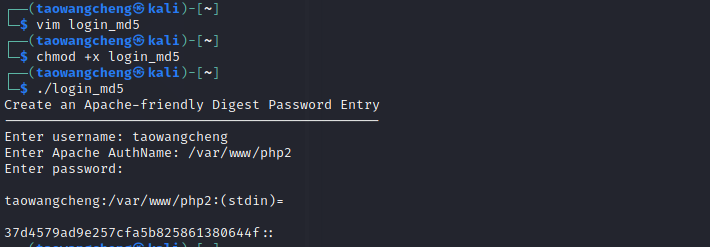


图5-6 运行脚本的截图

* + 1. **关卡6对称密码**（Symmetric cryptography）

**14）对称密码加密/解密字符串**

先和你的同伴约定好通讯用的密码，再请你的同伴，利用openssl命令加密后发送到课程群，@你；你尝试解密这条信息，并记录结果；

**解答：**

（a）使用AES128算法，执行命令echo "Hello, WangZhuo, I'm

TaoWangcheng, using openssl here" | openssl enc -aes128 -e -a -salt生成加密信息，其中加密口令为hust123，如图6-1所示。

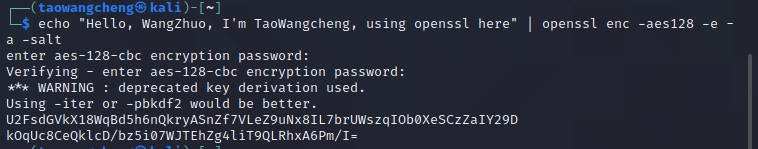


图6-1 对称密码加密字符串的截图

（b）接收同伴的加密信息，执行命令echo

"U2FsdGVkX1/BjIKtq4r909ke4tYxY/3l25UPEzp/YFs2wzIKE59ZSg096SUmRLWN

0cc9K/5rIAubUhBA3N0dAGhjE7urd7p7nuNggeHZ/8E=" | openssl enc -aes128 -d -a解密信息，其中加密口令为hust123，如图6-2所示。

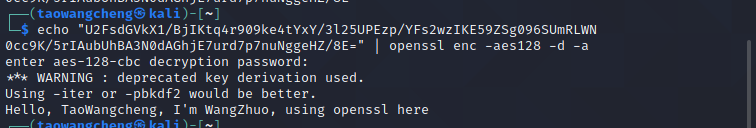


图6-2 对称密码解密字符串的截图

**15）加密文本**

与小任务14类似，但换一种加密算法参数，传递一个简短的文本文件：

**解答：**

（a）使用vim，编辑文本文件，如图6-3所示。

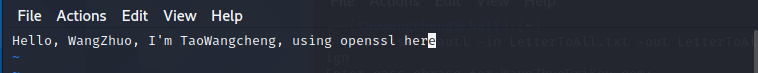


图6-3 编辑文本文件的截图

（b）执行命令生成密文，发给同伴，其中加密密钥为hust123，如图6-4所示。

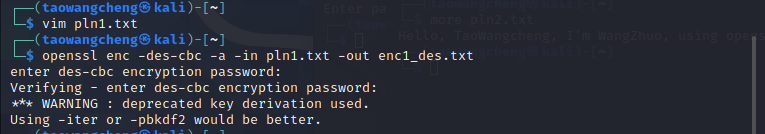


图6-4 生成密文的截图

（c）接收同伴的密文，执行命令解密密文，其中加密密钥为hust123，如图6-5所示。

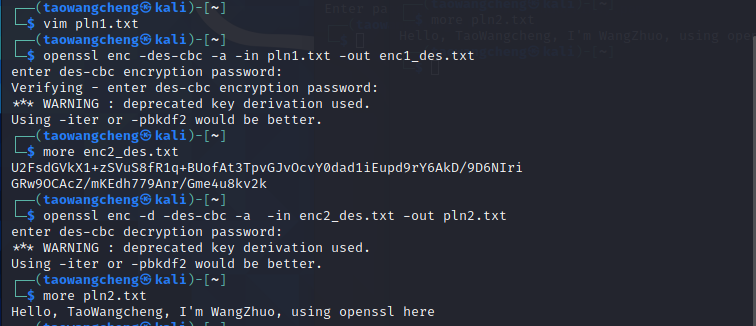


图6-5 解密密文的截图

* + 1. **关卡7非对称密码**（Asymmetric cryptography）

**16）生成私钥**

前面双方商量并交换password，实现起来安全吗？

回答：不安全，商量并交换password要建立在安全通信的基础上，而密码商议的过程容易被窃听。

公钥密码不需要密码商议，就能完成关卡8中的通讯。

首先，自己生成一个私钥，保存到文件中；

**解答：**

执行命令openssl genrsa -out TaoWangchengPriKey.pem 1024生成私钥，如图7-1所示。

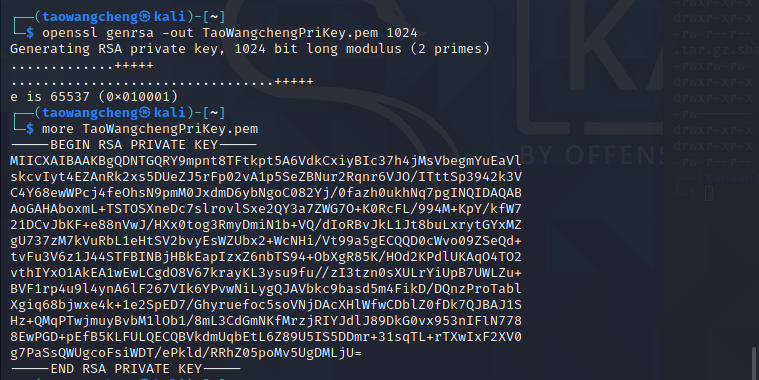


图7-1 生成私钥的截图

**17）非对称密码加密文件**

利用openssl rsautl加密一个文件。

**解答：**

执行命令openssl rsautl -encrypt -in test.txt -out enc.txt -inkey TaoWangchengPriKey.pem加密文件test.txt并输出到enc.txt中，如图7-2所示。

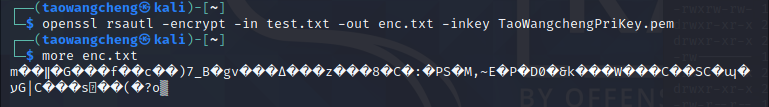


图7-2 非对称密码加密文件的截图

**18）非对称密码解密文件：**

利用openssl rsautl解密一个文件。

**解答：**

执行命令openssl rsautl -decrypt -in enc.txt -out temp.txt -inkey TaoWangchengPriKey.pem解密文件enc.txt并输出到temp.txt中，如图7-3所示。

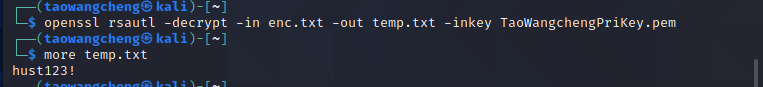


图7-3 非对称密码解密文件的截图

**19）保护密钥**

现在的问题，你的密钥文件包含的私钥，没有任何保护，任何人如果取得，都可以使用；所以需要加密存放密钥文件本身。请生成rsa私钥并选一种密码算法，保护你的私钥。可以选DES、DES3、IDEA、AES128、AES192、AES256。

**解答：**

采用DES密码算法，执行命令openssl genrsa -out

TaoWangchengPriKey2.pem -des 1024生成RSA私钥并加密，其中加密口令为hust123，如图7-4所示。

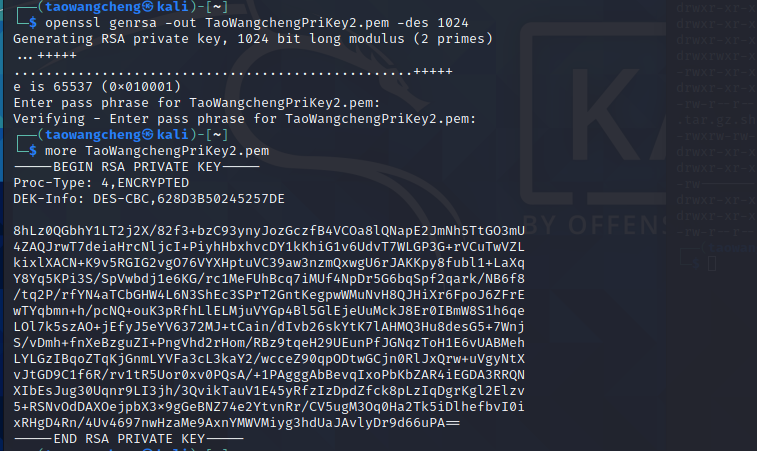


图7-4 生成RSA私钥并加密的截图

* + 1. **关卡8 星际通讯**

公司派你出差到了芝加哥，你需要与你在Galaxy公司的伙伴Bob（你的同组人），进行加密通讯，你们将公钥通过钉钉群，公开传递，互换公钥，之后用对方的公钥，加密你们要交换的文件。并检查结果是否正确。

**20）分离公钥**

同组人，分别分离出自己的公钥，保存到自己的文件中，发到课程群内，并保存另一方的公钥文件；显示自己公钥的内容；

**解答：**

（a）执行命令openssl rsa -in TaoWangchengPriKey2.pem -pubout -out TaoWangchengPubKey.pem分离公钥，其中输入的口令为hust123，如图8-1所示。

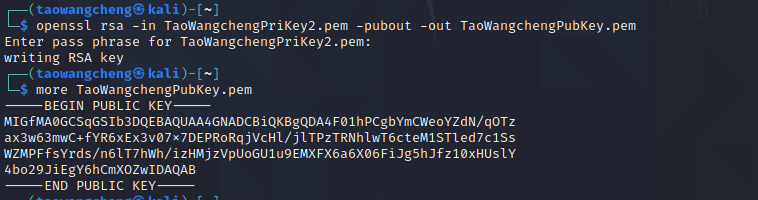


图8-1 用openssl分离公钥的截图

（b）保存同伴的公钥文件，如图8-2所示。

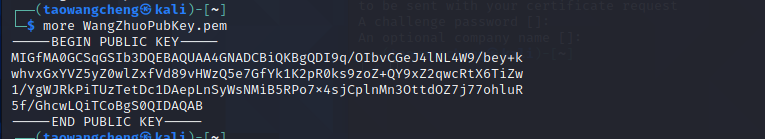


图8-2 查看同伴公钥的截图

**21）加密邮件**

准备一封写给对方的信，然后使用对方的公钥，加密信件，并通过邮件和课程群，公开发出给同组伙伴Bob；

**解答：**

（a）使用vim，准备写给对方的信，如图8-3所示。

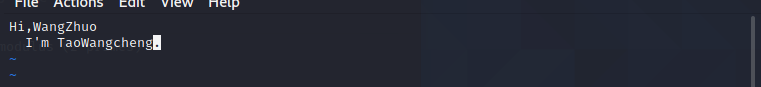


图8-3 编辑信件的截图

（b）执行命令openssl rsautl -encrypt -pubin -inkey

WangZhuoPubKey.pem -in LetterToWangZhuoFromTaoWangcheng.txt -out

LetterToWangZhuoFromTaoWangchengEnc.txt加密信件，如图8-4所示。

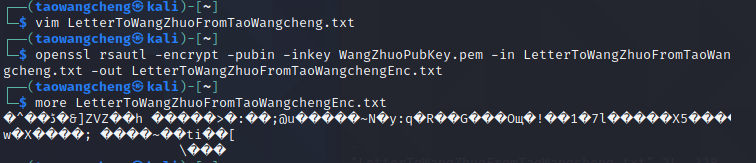


图8-4 加密信件的截图

**22）私钥解密**

你公开获得密文邮件后，能解密出你的同伴给你发的内容，请记录：

**解答：**

执行命令openssl rsautl -decrypt -inkey TaoWangchengPriKey2.pem -in LetterToTaoWangchengFromWangZhuoEnc.txt -out

LetterToTaoWangchengFromWangZhuo.txt解密信件，其中输入口令为hust123，并查看解密后的信件内容，如图8-5所示。

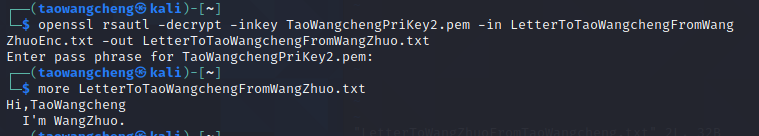


图8-5 解密信件的截图

**23）团队合作**

你的伙伴能还原出你发给他的加密邮件吗？

记录： 能，如图8-6所示。



图8-6 同伴解密成功的截图

* + 1. **关卡9 May The Cryptography Force Be With U**

**24)数字签名**

假如你担任了华东市场的经理，发一封邮件给你的团队：“Dear team members, the most difficult time is about to pass, with a bright future---：XXXXX”。员工Bob收到后，如何能确认邮件是你发出的？

同组两位同学，互相验证对方的签名信件，并记录信件的内容：

**解答：**

（a）使用vim，编辑邮件，如图9-1所示。

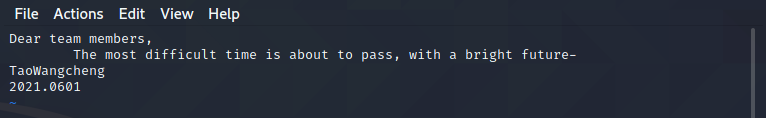


图9-1 编辑邮件的截图

（b）执行命令openssl rsautl -in LetterToAll.txt -out

LetterToAllSig.txt -inkey TaoWangchengPriKey2.pem -sign生成自己的签名信件，其中输入的口令为hust123，并发送给同伴，如图9-2所示。

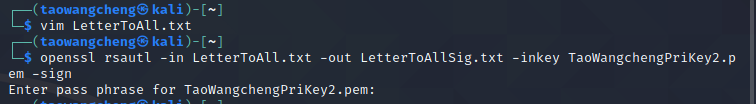


图9-2 生成自己签名信件的截图

（c）接收同伴的加密信件，执行命令openssl rsautl -in

LetterToAllSig.txt -inkey WangZhuoPubKey.pem -pubin -verify验证同伴的签名信件，如图9-3所示。

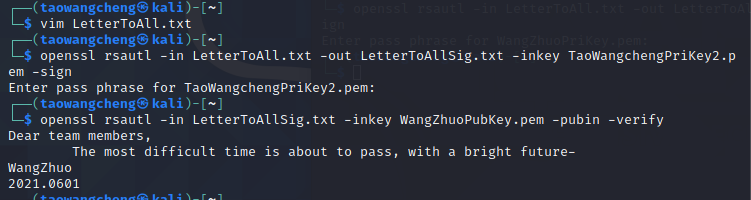


图9-3 验证同伴签名信件的截图

* + 1. **关卡10 待下回分解To be continued**

**25)请求证书**

你毕业后，准备成立自己的初创公司，得取一个响亮的名字吧，比如：Universe，给自己的公司网站签发证书。你的公司可能也需要一个密钥对。请利用openssl生成公司的私钥，另外，完成签发证书的请求。

**解答：**

（a）先执行命令openssl genrsa -out UniversePriKey.pem -des3 2048生成私钥，如图10-1所示。

（b）再执行命令openssl req -new -key UniversePriKey.pem -out Request.pem完成签发证书的请求，并按提示填写信息，如图10-1所示。

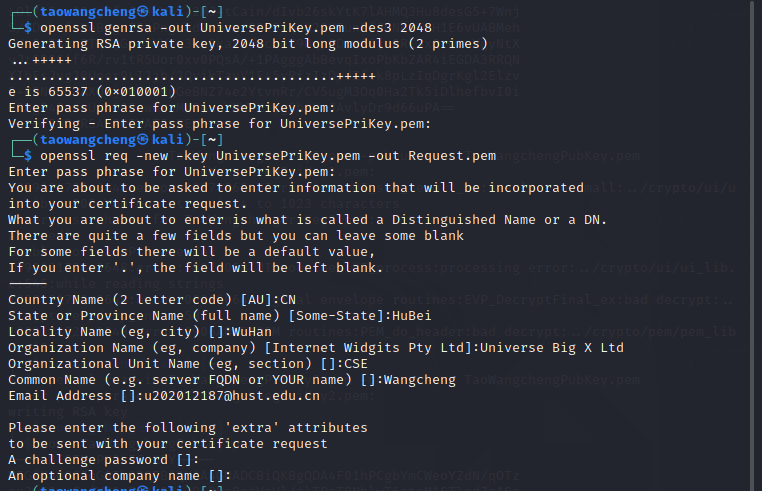


图10-1 请求证书的截图

* + 1. **扩展阅读及参考**

1. <Https://luv.asn.au/overheads/security/index.html>
2. [OpenSSL与网络信息安全-基础、结构和指令.pdf](file:///E:\2020分级培养一\2%20加密解密（GnuPG、Hashcat、advanced%20archive%20password%20recovery）\OpenSSL与网络信息安全-基础、结构和指令.pdf)
3. [Openssl man1.pdf](file:///E:\2020分级培养一\2%20加密解密（GnuPG、Hashcat、advanced%20archive%20password%20recovery）\openssl%20man1.pdf)
4. [Openssl开发手册.chm](file:///E:\2020分级培养一\2%20加密解密（GnuPG、Hashcat、advanced%20archive%20password%20recovery）\openssl开发手册.chm)
5. [Hashcat.pdf](file:///E:\2020分级培养一\2%20加密解密（GnuPG、Hashcat、advanced%20archive%20password%20recovery）\hashcat.pdf)
6. [Crunch.pdf](file:///E:\2020分级培养一\2%20加密解密（GnuPG、Hashcat、advanced%20archive%20password%20recovery）\crunch.pdf)
7. <Http://www.linuxguruz.com/forum/security-f554.html>

\***可选关卡**：

1、Openssl里面有没有我们国家自己的标准算法？

2、暴力破解关卡中，能否提高破解速度？

**解答：**

（1）Openssl里面有我们国家自己的标准算法，例如SM系列的sm4算法。

（2）可以通过已知的信息缩小范围、优化虚拟机的配置（如基于GPU）以及分布式破解等方法来提高暴力破解的速度。

## 实验问题分析与总结

## 关键分析

1. RAR文件破解带来的思考：

不要轻易缩小密码可选择的范围，会导致破解难度指数级降低。从暴力破解提示前的难度和暴力破解提示后的难度的对比可以看出。

（2）用户密码加密算法

旧版本Linux使用SHA512加密用户密码，新版Linux采用yescrypt hash加密，而这可能导致无法用hashcat破解，详细原因需要进一步研究，可参考附件。

（3）密码破解

分别通过Windows图形界面、Linux命令行工具学习了密码破解的内容，其中包括缩小范围暴力破解、字典破解，更多内容需要进一步实践学习。

（4）MD5与AES加密破解

MD5字段加密可以用hashcat破解而AES加密字段不能用hashcat破解，原因在于MD5加密采用的是密码散列函数，通过利用其性质：

“找到任意两个不同的报文x和y使得H(x)=H(y)，在计算上是不可能的。”

可以实现暴力破解。而AES加密除了未知的明文外，还有一个未知的加密口令，导致无法用hashcat破解。

（5）Shell编程

在模拟登录的背景下，编写了一个简单的Shell脚本。

（6）对称密码与非对称密码

通过实际情景下的安全通信，进一步加深了对于对称密钥系统和公开密钥系统的理解。主要区别在于对称密钥系统中，Alice和Bob的密钥是相同的且是秘密的，而公开密钥系统中，使用一对密钥：一个密钥为Bob和Alice共知（实际为全世界所知），即公钥；另一个密钥为Bob或Alice一方知道，即私钥。具体分别实践了对于文件的加密与解密操作。

深入到公开密钥加密与解密中，其本质为K+(K-(m))=K-(K+(m))=m（m为明文，K+为公钥，K-为私钥），在此基础上实现了文件的加密与解密、数字签名和公钥认证等操作。

值得一提的是本次实验中的数字签名采用的是完全加密/解密对数据签名，更为有效的一种方法是，将散列函数引入数字签名：使用散列函数，对报文的散列签名而不是对报文本身签名，生成数字签名所需要的计算量大为降低。

## 实验小结

本次实验主要学习加密与解密的相关知识和操作，激发学习密码学的兴趣。首先学习了利用工具实现密码破解的操作，接着分别在对称密钥系统和公开密钥系统中实现了安全通信的一些基本操作。密码学技术在网络安全中的地位十分重要，与鉴别、报文完整性和不可否认性密切相关，也用于生成安全的网络协议。对密码学的全面讨论需要一本完整的书，需要在今后的密码学课程中进一步学习。

## 参考文献及资料列表

[1]James F. Kurose,Keith W.Ross.Computer Networking: A Top-Down Approach

(Seventh Edition).北京:机械工业出版社,2018,5

[2]<https://www.cnblogs.com/diligenceday/p/6359661.html>

[3]<https://blog.csdn.net/SHIGUANGTUJING/article/details/90074614>