Security-02

服务安全与监控

# 加密与解密

## 加/解密概述

### 信息传递中的风险：

电脑黑客、出差人员、离职人员、合作伙伴、商业间谍、高管习惯、开发人员、流程失控

### 什么是加/解密

发送方：明文——>密文

接收方：密文——>明文

### 加密目的及方式

### 1. 确保数据的机密性

- 对称加密：加密/解密用通一个密钥

- 非对称加密：加密/解密用不同的密钥（公钥、私钥）

### 2. 保护信息的完整性

信息摘要：基于输入的信息生成长度较短、位数固定的散列值

### 常见的加密算法

1. 对称加密

- DES, Data Encryption Standard

- AES, Advanced Encryption Standard

1. 非对称加密

- RSA , Rivest Shamirh Adleman

- DSA , Digital Signature Algorithm

1. Hash散列技术，用于信息摘要

- MD5 , Message Digest Algorithm 5

- SHA , Secure Hash Algorithm

根据输入的文本（长度不限），生成固定长度（比如123位）的摘要文本；只要输入的文本不同，则生成的摘要文本也不一样

### MD5完整性检验

使用md5sum校验工具

- 生成MD5检验值

- 与软件官方提供的检验值比对

[root@client50 ~]# md5sum pass.txt

aac75645117ce68ce0870788824adaec pass.txt

**[root@client50 ~]# ifconfig eth0 | md5sum**

6868bd29c5c535df911473b66b813c30 -

# GnuPG加/解密工具

## GnuPG简介

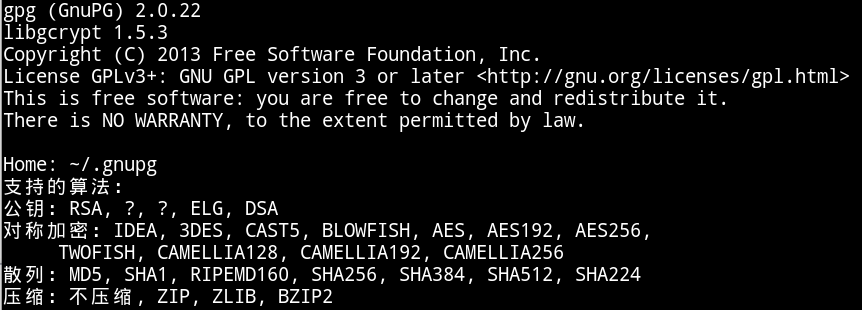
### GnuPG, GNU Privacy Guard

- <http://www.gnupg.org/>

- 最流行的数据加密、数字签名工具软件

### 安装gnupg2-2.0.22-4.el7.x86\_64软件包提供gpg命令（默认是安装的）

[root@client50 ~]# gpg --version



## GPG对称加/解密

### 测试环境准备：

创建两个用户usera和userb

[root@client50 ~]# useradd usera

[root@client50 ~]# useradd userb

[root@client50 ~]# echo 123456 | passwd --stdin usera

[root@client50 ~]# echo 123456 | passwd --stdin userb

### 基本用法

- 加密操作： --symmetric或 -c

- 解密操作： --decrypt 或 -d

### 发送方加密文件

[root@client50 ~]# ssh -X usera@localhost

[usera@client50 ~]$ cat a.txt

abc

123

bcd

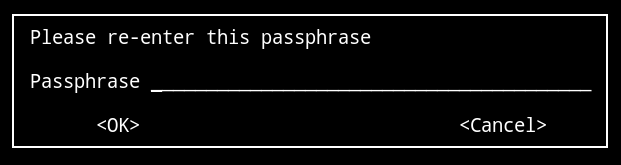
456

[usera@client50 ~]$ gpg -c a.txt

输入加密密码



确认密码



[usera@client50 ~]$ ls

a.txt a.txt.gpg cluster.tar.gz //加密后的文件

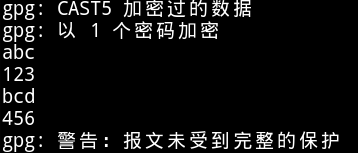
[usera@client50 ~]$ cp a.txt.gpg /tmp

### 接收方解密文件

新开终端利用userb用户连接

[root@client50 ~]# ssh -X userb@localhost

[userb@client50 ~]$ gpg -d /tmp/a.txt.gpg

//输入的密码与发送方加密时的密码一致,解密之后即可看到加密文件的内容

## GPG非对称加/解密

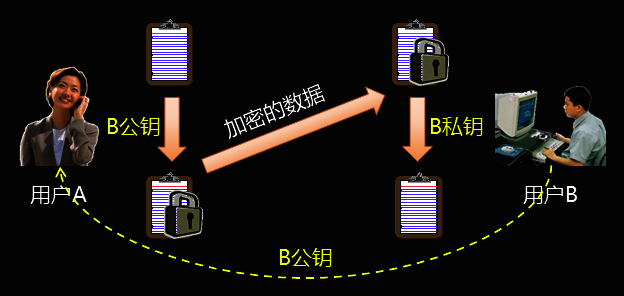
usera[接收方]

创建密钥对——>导出公钥——>把公钥文件传给用户userb

userb[发送方]

**导入公钥——>加密明文文件——>把加密后的文件传给用户usera——>用户usera解密文件**

### 基本过程



01：用户B把自己的公钥传给用户A

02：用户A使用B的公钥加密数据，将加密后的数据给用户B

03：用户B使用自己的私钥解密数据

## 准备：

userb创建密钥对： --gen-key

userb导出公钥： --export、--armor 或 -a

usera 导入公钥： --import

### 创建密钥对：

[usera@client50 ~]$ rm -rf ~/.pnupg

[usera@client50 ~]$ gpg --gen-key

.. ...

请选择您要使用的密钥种类：

(1) RSA and RSA (default)

(2) DSA and Elgamal

(3) DSA (仅用于签名)

(4) RSA (仅用于签名)

**您的选择？ //直接回车选择默认**

RSA 密钥长度应在 1024 位与 4096 位之间。

**您想要用多大的密钥尺寸？(2048) 1024 //根据自己选择填写**

您所要求的密钥尺寸是 1024 位

请设定这把密钥的有效期限。

0 = 密钥永不过期

<n> = 密钥在 n 天后过期

<n>w = 密钥在 n 周后过期

<n>m = 密钥在 n 月后过期

<n>y = 密钥在 n 年后过期

**密钥的有效期限是？(0) //直接回车选择默认**

密钥永远不会过期

**以上正确吗？(y/n)y //输入y确认**

You need a user ID to identify your key; the software constructs the user ID

from the Real Name, Comment and Email Address in this form:

"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

**真实姓名：asdqwe //名字不少于5位（记住后面有用）**

**电子邮件地址：[asd\_qwe@sina.cn](mailto:asd_qwe@sina.cn) //输入你的邮箱**

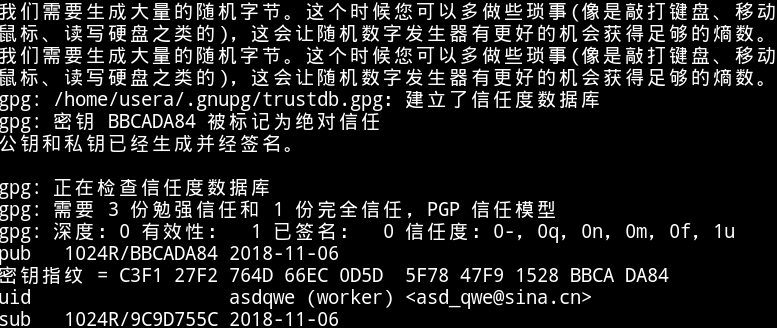
**注释：worker //描述信息随机写**

您选定了这个用户标识：

“asdqwe (worker) <asd\_qwe@sina.cn>”

**更改姓名(N)、注释(C)、电子邮件地址(E)或确定(O)/退出(Q)？O //输入大写O确认**

**您需要一个密码来保护您的私钥。 //根据提示输入密码和确认密码**

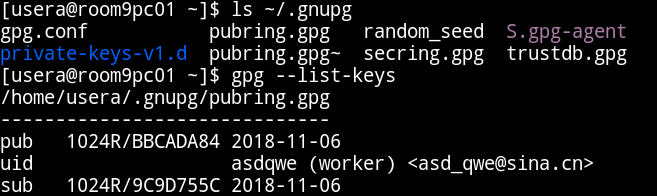


如果生成不了，可以试试向硬盘中写数据，执行此命令：

[root@client50 ~]# dd if=/dev/zero of=/a.txt bs=1M count=10240

[usera@client50 ~]$ ls ~/.gnupg

[usera@client50 ~]$ gpg --list-keys

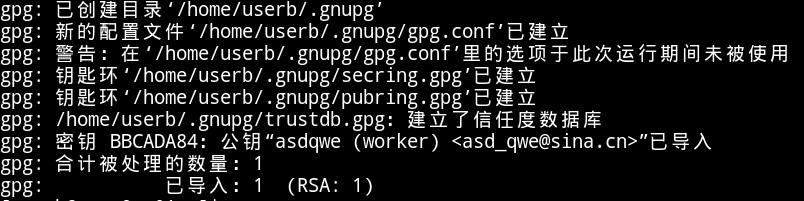


导出公钥

[usera@client50 ~]$ gpg --export -a > /tmp/userb.pub

userb 导入公钥

[userb@client50 ~]$ gpg --import /tmp/userb.pub



### 基本用法

加密操作：--encrypt 或 -e;;

指定目标用户: --recipient 或 -r

解密操作： --decrypt 或 -d

userb根据usera传给自己的公钥对文件进行加密

[userb@client50 ~]$ echo qwertyuiasdffghghjh > t2.txt

[userb@client50 ~]$ gpg -e -r asdqwe t2.txt //加密，根据提示选择

[userb@client50 ~]$ cp t2.txt.gpg /tmp/

usera进行解密

[usera@client50 ~]$ gpg -d /tmp/a.txt.gpg > b.txt

//解密，按要求输入密码

[usera@client50 ~]$ cat b.txt

## GPG软件签名与验证

### 软件签名与验证过程

软件官方以私钥对软件包执行数字签名

用户下载软件包，软件官方的公钥

以官方公钥验证软件包签名，确保数据来源正确

### 为软件包建立签名文件

分离式签名： **--detach-sign 或 -b**

### 验证软件包签名

验证签名：**--verify**

在usera上进行签名认证，在userb上进行签名验证

[usera@client50 ~]$ cat b.txt

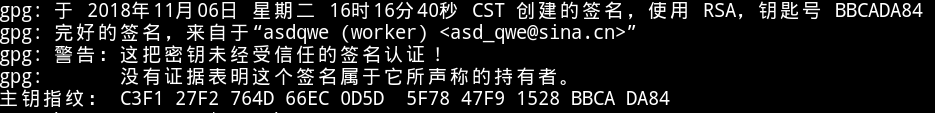
[usera@client50 ~]$ gpg -b b.txt //对文件进行签名认证

[usera@client50 ~]$ cp b.txt\* /tmp/

//将源文件与签名认证文件传给userb

[userb@client50 ~]$ gpg --import /tmp/userb.pub

[userb@client50 ~]$ gpg --verify /tmp/b.txt.sig //验证签名



# AIDE入侵检测系统

## 初始化系统

### 安装软件包

AIDE(Advanced intrusion deteciton environment)

该软件为一套入侵检测系统

配置yum源即可安装aide软件

[root@client50 ~]# yum -y install aide

### 修改配置文件

AIDE默认配置文件为/etc/aide.conf

### 初始化检查

在没有被攻击入侵前

根据配置文件，对数据进行校验操作

[root@client50 ~]# cp /etc/aide.conf{,.bak}

[root@client50 ~]# vim /etc/aide.conf

3 @@define DBDIR /var/lib/aide //数据库目录

4 @@define LOGDIR /var/log/aide //日志目录

12 database\_out=file:@@{DBDIR}/aide.db.new.gz //数据库文件名

15 gzip\_dbout=yes //压缩输出

28 #p: permissions //希望检查的项目

29 #i: inode:

30 #n: number of links

31 #u: user

32 #g: group

33 #s: size

... ....

42 #md5: md5 checksum

43 #sha1: sha1 checksum

44 #sha256: sha256 checksum

... ....

54 FIPSR = p+i+n+u+g+s+m+c+acl+selinux+xattrs+sha256

98 /testdir FIPSR //自行添加

99 /boot/ CONTENT\_EX //对那些目录进行什么校验

100 /bin/ CONTENT\_EX

101 /sbin/ CONTENT\_EX

102 /lib/ CONTENT\_EX

103 /lib64/ CONTENT\_EX

104 /opt/ CONTENT

112 !/usr/src/ //使用[!]，设置不校验的目录

113 !/usr/tmp/

：99，312s/^/#/ //将99-312行全部注释

[root@client50 ~]# mkdir /testdir

[root@client50 ~]# vim /testdir/a.txt

[root@client50 ~]# aide --init //生成监视目录的初始目录

AIDE, version 0.15.1

### AIDE database at /var/lib/aide/aide.db.new.gz initialized.

### 备份数据库

再被入侵前，将校验的数据库文件备份到安全的地方

如：U盘，光盘、移动硬盘、网络存储

[root@client50 ~]# cp /var/lib/aide/aide.db.new.gz /root/

//对初始目录进行备份

## 入侵检查

### 执行入侵检查

将之前备份的校验数据库文件还原

[root@client50 ~]# mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz

根据数据库执行入侵检测

[root@client50 ~]# vim /testdir/a.txt

[root@client50 ~]# mkdir /testdir/b

[root@client50 ~]# aide -C(check)

# 扫描与抓包

## 为什么需要扫描

### 以获取一些公开/非公开信息为目的

- 检测前在的风险

- 查找可攻击目标

- 收集设备/主机/系统/软件信息

- 发现可利用的安全漏洞

## 扫描方式及工具

### 典型的扫描方式

Scan, 主动探测

Sniff, 被动监听/嗅探

Capture, 数据包捕获(抓包)

### 常见的安全分析工具

扫描器：NMAP

协议分析：tcpdump、WireShark

## NMAP简介

### 一款强大的网络探测里器工具

### 支持多种探测技术

ping扫描、多端口扫描、TCP/IP指纹校验

### 基本用法

nmap [扫描类型] [选项] <扫描目标...>

### 常用的扫描类型

-sS, TCP SYN扫描(半开)

-sT, TCP连接扫描(全开)

-sU, UDP扫描

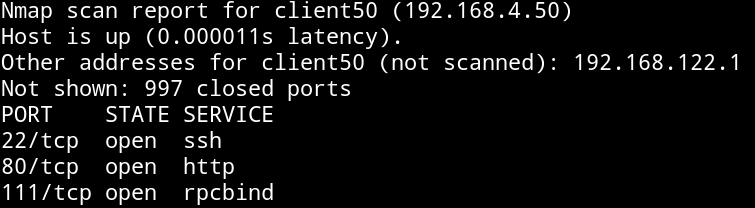
-sP, ICMP扫描

-A, 目标系统全面分析

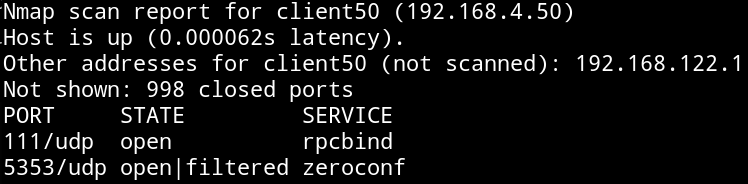
### 应用示例：

检查目标主机开放了那些端口

[root@client50 ~]# nmap client50 //默认扫描TCP

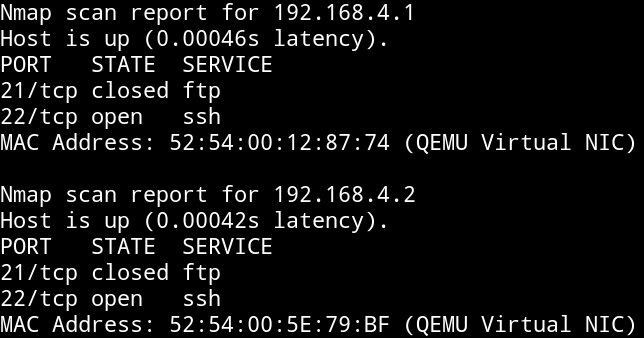


[root@client50 ~]# nmap -sU client50 //指定-sU扫描UDP



检查那些主机开启了FTP、SSH服务端

[root@client50 ~]# nmap -p 21-22 192.168.4.0/24



**//只要网段内能够连接的主机，都会有类似输出**

检查目标主机的存活状态（是否可ping通）

[root@client50 ~]# nmap -n -sP 192.168.4.0/24

Nmap scan report for 192.168.4.1

Host is up (0.00039s latency).

MAC Address: 52:54:00:12:87:74 (QEMU Virtual NIC)

... ...

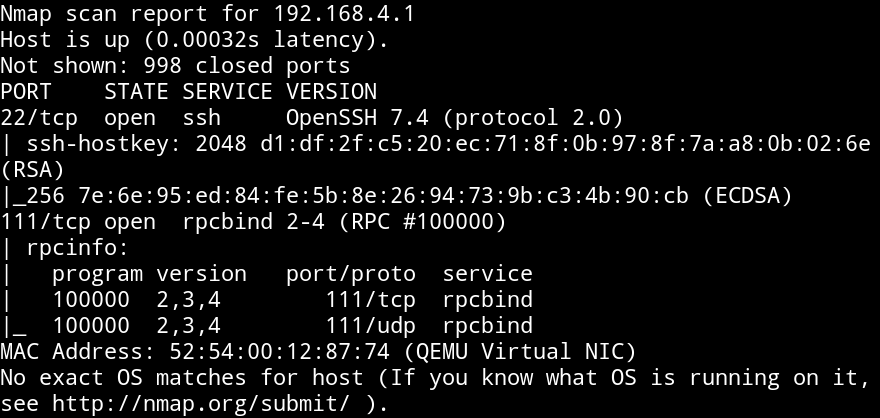
Nmap done: 256 IP addresses (6 hosts up) scanned in 1.44 seconds

检查操作系统指纹

-A:是一个复合选项，相当于：-O（OS检测）、-sV（版本检测）、-sC（脚本检测）、traceroute跟踪

[root@client50 ~]# nmap -A 192.168.4.1

... ...



... ...

# 网络抓包工具

## tcpdump抓包命令

一款提取TCP数据包的命令行工具(命令由软件tcpdump-4.9.0-5.el7.x86\_64提供)

### 基本用法

- tcpdump [选项] [过滤条件]

默认监控的网络接口为eth0网卡接口

### 常见监控选项

-i ,指定监控的网络接口

-A,转换为ACSII码，以方便阅读

-w,将数据包信息保存到指定文件

-r,从指定文件读取数据包信息

-c,指定抓包个数

]# man tcpdump

]# tcpdump -i enp3s0 -A -c 1 -w /tmp/tcpdump.cap

]# tcpdump -r /tmp/tcpdump.cap

### tcpdump的过滤条件

类型：host、net、port、portrange

方向：src(访问的方向)、dst(回应的方向)

协议：tcp、udp、ip、wlan、arp

多个条件组合：and、or、not

### ]# tcpdump -i eth0 -A -c 1 -w /tmp/ssh.cap tcp port 22

]# tcpdump -r /tmp/ssh.cap

]# tcpdump -i eth0 -A -c 1 tcp port 22 and host 192.168.4.58

//查看主机58是否在ssh本机

]# tcpdump -i eth0 -A -c 1 tcp port 80 and host 192.168.4.58

//查看主机58是否访问了本机的http服务

**根据访问的方向进行抓包**

]# tcpdump -A src host 192.168.4.58

10:24:18.294866 IP 192.168.4.58.41714 > client50.ftp: Flags [S], seq 1005197962, win 29200, options [mss 1460,sackOK,TS val 4124260 ecr 0,nop,wscale 7], length 0 //抓包信息显示58访问50的ftp服务

**根据回应的方向进行抓包**

]# tcpdump -A dst host 192.168.4.58

10:22:39.399939 IP client50.ftp > 192.168.4.58.41712: Flags [S.], seq 958042076, ack 1530800939, win 28960, options [mss 1460,sackOK,TS val 4037456 ecr 4025365,nop,wscale 7], length 0 //抓包信息显示50的ftp回复58

### 应用示例：

按条件（访问指定的POP3服务）抓取数据包

按CTRL+c键停止抓包

### 保存、分析抓包结果

- 抓取访问FTP服务的包，保存为cap文件

服务端]# useradd student

服务端]# echo 123456 | passwd --stdin student

服务端]# su - student

服务端]$ vim test.txt

服务端]$ exit

服务端]# tcpdump -i eth0 -A -w /tmp/ftp.cap tcp port 21

**客户端使用student用户连接50主机的ftp服务**

服务端]# tcpdump -r /tmp/ftp.cap

服务端]# tcpdump -A -r /tmp/ftp.cap | grep -i student

服务端]# tcpdump -A -r /tmp/ftp.cap | grep -i 123456

## WireShark协议分析器

一款与tcpdump类似的抓包工具，需要图形环境

### - <http://www.wireshark.org/>

### RHEL光盘中的2个包

- wireshark

- wireshark-gnome

安装软件包：

[root@room9pc01 ~]# yum -y install wireshark wireshark-gnome