12_security04 加密与解密+AIDE 入侵检测系统+扫描与抓包

一 加密与解密概述

1.1 信息传递中的风险:技术手段风险和非技术手段风险

1.2 什么是加/解密

加密:把明文变成密文的过程,发送方发负责加密

解密:把密文变成明文的过程,接收方负责解密

1.3 加密目的及方式

确保数据的机密性

对称加密:加密/解密用同一个密钥

非对称加密:加/解密用不同的密钥(公钥加密/私钥解密)

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

1.4 常见的加密算法

算法:数据加/解密时的计算规则

对称加密:

DES: Data Encryption Standard

AES: Advanced Encryption Standard

非对称加密:

RSA:Rivest Shamirh Adleman

DSA:Digital Signature Algorithm['ælgərɪðəm]

Hash 散列技术,用于信息摘要

MD5:Message Digest Algorithm 5

SHA: Secure Hash Algorithm

1.5 MD5/SHA 完整性校验

命令: md5sum/sha1sum /路径/文件 ls /路径/目录 | md5sum/sha1sum

二 GPG 加密与解密工具(真机登录 usera userb 两个账户时使用 ssh -X 方式)

2.1 GunPG 简介,系统默认安装

2.2 GPG 对称加/解密

命令: gpg 操作 /路径/文件

加密操作: --symmetric 或 -c

解密操作: --decrypt 或 -d

发送方加密文件

```
[usera@room9pc01 ~]$ vim test.txt
[usera@room9pc01 ~]$ gpg -c test.txt  #连续输入 2 次密码 1234567890
[usera@room9pc01 ~]$ mv test.txt.gpg /tmp/
```

接收方解密文件

```
[userb@room9pc01 tmp]$ cat test.txt.gpg  #打开后为乱码
[userb@room9pc01 tmp]$ gpg -d /tmp/test.txt.gpg > ~/a.txt
    #根据提示输入加密时的密码
```

2.3 GPG 非对称加/解密

公钥文件 加密

私钥文件 解密

2.3.1 b 创建密钥对 gpg --gen-key

```
[userb@room9pc01 ~]$ gpg --gen-key  #根据提示设置信息
```

1->2048->0->y->真实姓名->电子邮件->注释->大写 O->弹出的窗口输入私钥保

护密码两次:1234567890->

```
[userb@room9pc01 ~]$ ls -a ~/.gnupg/
```

#重新创建密钥对时删除此目录下所有内容

生成密钥排错

```
mv /dev/random /dev/randomold
ln -s /dev/urandom /dev/random
```

2.3.2 b 导出公钥 **gpg --export --armor** 或**-a** **#默认输出到屏幕上**

```
[userb@room9pc01 ~]$ gpg --export -a > userb.pub
[userb@room9pc01 ~]$ mv userb.pub /tmp/
```

2.3.3 a 导入公钥 **gpg --import /路径/公钥文件名**

```
[usera@room9pc01 ~]$ gpg --import /tmp/userb.pub
[usera@room9pc01 ~]$ ls ~/.gnupg/
```

#重新导入公钥时删除此目录下所有内容

2.3.4 a 用公钥加密文件 **gpg --encrypt** 或**-e -r** 用户名 **/路径/**文件名

```
[usera@room9pc01 ~]$ gpg -e -r userb 1.txt  #提示信息输入 y
[usera@room9pc01 ~]$ ls
```

```
1.txt  1.txt.gpg    #多出了 1.txt.gpg 这个加密文件
```

2.3.5 a 共享加密的文件

```
[usera@room9pc01 ~]$ mv 1.txt.gpg /tmp/
```

2.3.6 b 使用私钥解密文件 gpg --decrypt 或 -d /路径/共享文件名

```
[userb@room9pc01 ~]$ gpg -d /tmp/1.txt.gpg > ~/1.txt
    #弹出窗口输入私钥保护密码
[userb@room9pc01 ~]$ cat 1.txt
```

2.4 GPG 软件签名与验证(验证文件的完整性)

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+软件官方公钥->

以官方公钥验证软件包签名,确保数据来源正确

2.4.1 软件包建立签名文件 gpg --detach-sign 或 -b /目录/文件名

userb 用私钥对软件包进行签名并共享

```
[userb@room9pc01 ~]$ cat a.txt
i love you china
[userb@room9pc01 ~]$ gpg -b ~/a.txt   #输入私钥保护密码
[userb@room9pc01 ~]$ ls  #多了一个 a.txt.sig 签名文件
```

```
[userb@room9pc01 ~]$ mv a.txt a.txt.sig /tmp
```

创建公钥并共享

```
[userb@room9pc01 ~]$ gpg --export -a > userb.pub
[userb@room9pc01 ~]$ mv userb.pub /tmp/
```

2.4.2 验证软件包签名 gpg --verify /目录/签名文件名

```
[usera@room9pc01 ~]$ gpg --verify /tmp/a.txt.sig
```

三 AIDE 入侵检测系统-初始化系统

Advanced Intrusion[ɪn'truːʒn] Detection Environment

该软件为一套入侵检测系统,配置 yum 源即可安装 aide 软件,只能检查,不能阻止

3.1 安装软件包

```
svr7 ~]# yum -y install aide
```

3.2 修改配置文件 /etc/aide.conf

注释/etc//aide.conf 99 行以后的内容

```
svr7 ~]# sed -i '99,312s/^/#/' /etc/aide.conf
svr7 ~]# vim +98 /etc/aide.conf
```

添加以下内容

```
98 /root/  FIPSR    #54 行对 FIPSR 做了规则定义,对/root/目录进行检测
```

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

3.3 初始化检查,生成初始信息

```
svr7 ~]# ls /var/lib/aide    #为空
svr7 ~]# aide --init    #初始化
svr7 ~]# ls /var/lib/aide
aide.db.new.gz    #被检测目录/root/未被入侵前的初始信息
```

3.4 备份数据初始信息

```
svr7 ~]# mv /var/lib/aide/aide.db.new.gz /opt/  #先移出,再复制回来
svr7 ~]# cp /opt/aide.db.new.gz /var/lib/aide/aide.db.gz
  #配置文件中定义的文件名,系统将操作写入 aide.db.gz,然后与
  aide.db.new.gz 做比较,不同则有入侵
```

3.5 执行入侵检查 aide --check

```
svr7 ~]# aide --check  #根据输出信息,可查看/root/目录的修改信息
```

四 扫描与抓包-安全分析概述

4.1 为什么需要扫描

以获取一些公开/非公开信息为目的

检查潜在的风险

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

4.2 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)

五 NMAP 扫描

5.1 NMAP 简介

一款强大的网络探测利器工具

支持多种探测技术

ping 扫描

多端口扫描

TCP/IP 指纹校验

.....

常见的安全分析工具,扫描器:nmap,协议分析:tcpdump wireshark

5.2 NMAP 应用示例

基本用法

```
nmap [扫描类型] [选项] <扫描目标...>
```

常用扫描类型

-sS, TCP SYN 扫描(半开)

#sacn tcp 扫描主机连接被扫描主机时只有前 2 次握手,不向被扫描主机提供第 3 次握手的确认信息

-sT, TCP 连接扫描(全开)

#scan tcp 扫描主机连接被扫描主机时有 3 次握手

-sU, UDP 扫描 #scan udp

-sP, ICMP 扫描 #scan ping

常用扫描选项

-A, 目标系统全面分析

-p, 指定扫描端口

-n, 不做域名解析,使扫描速度更快

扫描网段内开机的主机

```
[root@room9pc01 ~]# nmap -sP 176.204.27.0/24
```

扫描网段内开机的主机开启的服务

```
[root@room9pc01 ~]# nmap 176.204.27.162
```

全面扫描某台主机

```
[root@room9pc01 ~]# nmap -A 176.204.27.162
```

单独扫描某些主机

```
[root@room9pc01 ~]# nmap -sP 176.204.27.162
[root@room9pc01 ~]# nmap -sP 176.204.27.100-200
[root@room9pc01 ~]# nmap -sP 176.204.27.100,110,120,130
```

端口扫描

```
[root@room9pc01 ~]# nmap -p 80 176.204.27.162
[root@room9pc01 ~]# nmap -p 80,21,22 176.204.27.162
[root@room9pc01 ~]# nmap -p 20-25 176.204.27.162
```

不作域名解析扫描,扫描速度更快

```
[root@room9pc01 ~]# nmap -n -p 20-25 176.204.27.162
```

其他命令: man nmap

六 网络抓包工具

前提:数据要经过抓包主机

6.1 tcpdump 抓包命令

tcpdump 抓包命令

tcpdump [选项] [过滤条件]

常见选项

-i 指定监控的网络接口

-A 转换为**ASCII**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 定义抓包个数

6.2 tcpdump过滤条件

类型: **host\net\prot\partrange**

方向: **src(来自于)\dst(去向)**

协议: **tcp\udp\ip\wlan\arp ...**

多个条件组合: **and\or\not**

ctrl+c 停止抓包

带选项案例

```
[root@room9pc01 ~]# tcpdump -i eth0
[root@room9pc01 ~]# tcpdump -i eth0 -c 2
[root@room9pc01 ~]# tcpdump -i eth0 -c 2 -w /tmp/a.cap
[root@room9pc01 ~]# tcpdump -r /tmp/a.cap
[root@room9pc01 ~]# tcpdump -A -i eth0 -c 2 -w /tmp/b.cap
[root@room9pc01 ~]# tcpdump -A -r /tmp/b.cap
```

带过滤条件案例

```
[root@room9pc01 ~]# tcpdump -i eth0 host 176.204.27.162
[student@room9pc01 ~]$ ping 176.204.27.162
```

```
[root@room9pc01 ~]# tcpdump -i eth0 dst host 176.204.27.181
[student@room9pc01 ~]$ ping 176.204.27.181

[root@room9pc01 ~]# tcpdump -i eth0 src host 176.204.27.162
[student@room9pc01 ~]$ ping 176.204.27.162

[root@room9pc01 ~]# tcpdump -i eth0 icmp and host 176.204.27.181
[student@room9pc01 ~]$ ping 176.204.27.181

[root@room9pc01 ~]# tcpdump -i eth0 icmp or host 176.204.27.181
[student@room9pc01 ~]$ ping 176.204.27.181
```

6.3 保存\分析抓包结果

6.3.1 抓取访问 **FTP** 服务的包,保存为 **cap** 文件

```
[root@srv7 ~]# useradd zhangsan
[root@srv7 ~]# echo 123456 | passwd --stdin zhangsan
[root@srv7 ~]# tcpdump -i eth0 tcp and \(port 21 or port 20\)

[root@svr8 ~]# yum -y install ftp  #安装登录 ftp 软件
```

[root@svr8 ~]# ftp 192.168.4.7 #用户密码分别是 zhangsan 和 123456

[root@srv7 ~]# tcpdump -A -w ftp.cap host 192.168.4.8 and \(port 21 or port 22\)

[root@svr8 ~]# ftp 192.168.4.7 #用户密码分别是 zhangsan 和 123456

6.3.2 分析抓取结果

tcpdump -A -r ftp.cap | egrep '(USER|PASS)'

09:52:14.574310 IP 192.168.4.8.37068 > ...,**USER zhangsan**

09:52:17.630055 IP 192.168.4.8.37068 >...**PASS 123456**

七 wireshark 协议分析器(服务器上应用较少)

7.1 wireshark 协议分析器概述

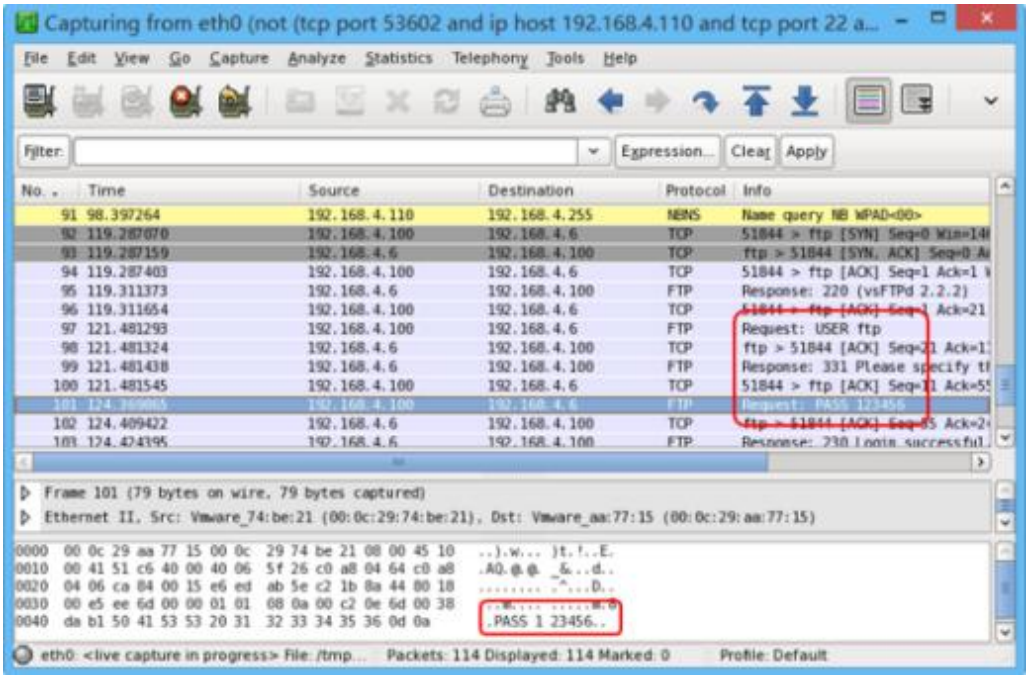
一款与 tcpdump 类似的抓包工具,需要图形环境,http://wwww.wireshakr.org

RHEL 光盘中的 2 个包: wireshark,wireshark-gnome

7.2 wireshark 界面

wireshark 抓包的效果如下图,与 tcpdump 类型,但需要图形

命令行输入:wireshark 打开应用程序图形界面



八 非对称加密原理

GPG 非对称加/解密