一、Nginx服务安全

P1 nginx安全

虚拟机提前安装Nginx ，修改nginx配置文件，开启autoindex关闭

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

server {

......

autoindex on;

}

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

移动网页目录下index.html，浏览器访问测试

[root@node1 ~]# cd /usr/local/nginx/html/

[root@node1 html]# mv index.html a.html

测试，会看到整个目录下所有资料



注释关闭autoindex，浏览器访问不会显示目录内容

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

server {

......

# autoindex on;

}

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

刷新网页，访问失败



所以在安装配置nginx之前可以加上 –without-http\_autoindex\_module禁用此模块

修改版本号信息

修改配置文件，http下添加server\_tokens off

重新加载配置，访问页面验证

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

http {

...

server\_tokens off;

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

Firefox 192.168.2.100 #刷新网页，此刻页面不显示版本号

限制并发



修改nginx配置文件，禁止过大的并发连接数

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

http {

...

limit\_req\_zone $binary\_remote\_addr zone=one:10m rate=1r/s; #nginx的内置变量，保存的远程我们的客户端IP，记录到内存里面，在内存里面开辟一个10M的空间，叫one，做限速，每秒一个请求

server {

...

limit\_req zone=one burst=5; #若发的请求多于1个，则放到缓冲的漏斗里面，可以处理5个请求

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

安装ab测试工具 测试验证

[root@node1 ~]# yum -y install httpd-tools

[root@node1 ~]# ab -c 100 -n 100 http://192.168.2.100/

......

Concurrency Level: 100

Time taken for tests: 5.002 seconds

Complete requests: 100 #发送100个并发连接数

Failed requests: 94 #失败的请求数

......

拒绝非法请求

node1编写网站首页，curl指定访问测试

[root@node1 ~]# echo hello > /usr/local/nginx/html/index.html

curl命令命令访问网站

[root@node1 html]# echo hello > /usr/local/nginx/html/index.html

[root@node1 html]# curl -i -X GET http://192.168.2.100/ #-i 指显示网页头部信息； -X(大写) 指定访问网站的方式

使用HEAD请求网页，只会获取网页头部信息，被攻击者所常用

[root@node1 ~]# curl -i -X HEAD http://192.168.2.100/

打开配置文件，server里任意位置进行修改

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

server {

......

if ($request\_method !~ ^(GET|POST)$ ) {

return 444;

}

# if条件判断，对于不是GET或PUT的所有请求，返回444错误

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

使用HEAD请求网页，直接返回错误信息

[root@node1 ~]# curl -i -X HEAD http://192.168.2.100/

curl: (52) Empty reply from server

防止buffer溢出

修改配置文件，重新加载配置文件

[root@node1 ~]# vim /usr/local/nginx/conf/nginx.conf

......

http {

client\_body\_buffer\_size 1k; #缓存用户的数据信息

client\_header\_buffer\_size 1k; #缓存用户的头部信息

client\_max\_body\_size 16k; #最大存储用户数据信息的缓存容量, 针对的是每一个用户的请求缓存的

large\_client\_header\_buffers 4 4k; #缓存用户的头部信息

......

重新加载配置文件

[root@node1 ~]# /usr/local/nginx/sbin/nginx -s reload

二、加密与解密

P1 GPG加/解密工具

MD5完整性检验

通过修改前后文件md5值变化，验证文件是否发生变化

[root@node1 ~]# md5sum test.txt

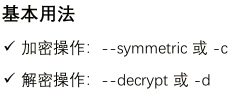
7d888795f59fbd130c5e251d7e53cf1c test.txt

[root@node1 ~]# vim test.txt #随意修改

[root@node1 ~]# md5sum test.txt #hash值会发生变化

c2c8b88bb1cfd53d3ad598a57511d5d6 test.txt

GPG加/解密工具



查看gpg 版本及算法

[root@node1 ~]# gpg --version

gpg (GnuPG) 2.0.22

…...

对称加密创建文件

进行文件加密，并查看

[root@web1 ~]# echo "hello the world" > test.txt

[root@web1 ~]# gpg -c test.txt # -c 加密文件，会提示输入两遍密码



删除原文件，进行解密查看加密文件

[root@node1 ~]# rm -rf test.txt

[root@node1 ~]# gpg -d test.txt.gpg #解密不需要输入密码，因为在加密的时候密码已经存放到文件中

如果在本机解密是需要输入密码，可以把.gnupg目录删除

[root@node1 ~]# rm -rf /root/.gnupg/

[root@node1 ~]# gpg -d test.txt.gpg

[root@node1 ~]# gpg -d test.txt.gpg > test.txt #解密的内容默认显示在屏幕上面，可以使用重定向放到文件里面

此时把这个加密文件传递给其他机器，如node2，解密时需要输入密码才行

[root@node1 ~]# scp test.txt.gpg 192.168.2.100:/root/

[root@node2 ~]# gpg -d test.txt.gpg #需要输入密码

非对称加/解密





node1主机生成密钥，公钥和私钥

如计算机长时间无法生成密钥 修改生成随机数

[root@node1 ~]# mv /dev/random /dev/random.bak

[root@node1 ~]# ln -s /dev/urandom /dev/random

[root@node1 ~]# gpg --gen-key #生成秘钥对

gpg (GnuPG) 2.0.22; Copyright (C) 2013 Free Software Foundation, Inc.

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

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？ #回车

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) #回车

您所要求的密钥尺寸是 2048 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) #回车

密钥永远不会过期

以上正确吗？(y/n)y #输入y

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：userb #秘钥名称

电子邮件地址：userb@tedu.cn #邮箱

注释：test key #注释，随意

您选定了这个用户标识：

“userb (test key) <userb@tedu.cn>”

更改姓名(N)、注释(C)、电子邮件地址(E)或确定(O)/退出(Q)？O #确定

您需要一个密码来保护您的私钥。 #直接回车，不给秘钥设置密码，选择take this one anyway，再次直接回车

生成后reset还原终端

导出密钥，并拷贝给node2

[root@node1 ~]# gpg --list-keys #查看密钥对

[root@node1 ~]# gpg -a --export userb > /tmp/userb.pub #导出秘钥，-a是导出ascii码的秘钥文件，如果不使用导出的是乱码的公钥文件

[root@node1 ~]# scp /tmp/userb.pub root@192.168.2.200:/tmp/ #传递给node2

node2导入公钥，并加密文件测试

[root@node2 ~]# gpg --list-keys

[root@node2 ~]# gpg --import /tmp/userb.pub

[root@node2 ~]# gpg --list-keys

node2主机生成明文数据，并加密

[root@node2 ~]# echo test > clear.txt

[root@node2 ~]# gpg -e -r userb clear.txt #gpg -e 加密数据，-r指定加密使用的公钥

Use this key anyway? (y/N) y #确认加密

[root@node2 ~]# scp clear.txt.gpg 192.168.2.100:/root/ #把加密数据传递给node1

node1解密查看文件

[root@node1 ~]#gpg -d -r userb clear.txt.gpg > clear.txt

[root@node1 ~]# cat clear.txt

test

GPG软件签名与验证

建立签名文件

[root@node1 ~]# gpg -b clear.txt

[root@node1 ~]# scp clear.txt clear.txt.sig 192.168.2.200:/root/

验证签名

[root@node2 ~]# gpg --verify clear.txt.sig clear.txt

三、系统审计

P1 部署audit

安装审计软件包，并启动服务

[root@node1 ~]# yum -y install audit

[root@node1 ~]# systemctl start auditd

[root@node1 ~]# systemctl enable auditd

P2 配置审计

查看审计状态，及审计规则

[root@node1 ~]# auditctl -s #查看状态

[root@node1 ~]# auditctl -l #查看规则

[root@node1 ~]# auditctl -D #删除规则

添加临时审计规则记录/etc/passwd文件变化

[root@node1 ~]# auditctl -w /etc/passwd -p rw -k passwd\_rule #-p后面加审计的行为，可以跟r,w,x,a(文件或者目录的属性发生变化)，-k给这条审计取的名称，便于识别

开启两个终端进行验证，一个终端查看文件，一个终端动态查看日志

[root@node1 ~]# tailf /var/log/audit/audit.log #终端二操作

[root@node1 ~]# cat /etc/passwd #终端一查看文件

[root@node1 ~]# useradd testuser2 #创建用户

添加新审计规则，记录parted命令

[root@node1 ~]# which parted

[root@node1 ~]# auditctl -w /usr/sbin/parted -p x -k parted\_disk

[root@node1 ~]# parted /dev/sda print #查看，此刻终端一即可看到日志

查看有关parted\_disk的日志信息。转化为易读的格式查看日志

[root@node1 ~]# ausearch -k parted\_disk -i #-i为交互式操作

补充：实现永久配置写入配置文件

[root@node1 ~]# vim /etc/audit/rules.d/audit.rules

-w /etc/passwd -p wa -k passwd\_rule