渗透测试思路

入侵渗透涉及许多知识和技术，下面简单概括下。

一，踩点

1.1 踩点可以了解目标主机和网络的一些基本的安全信息.

主要有：

1，管理员联系信息，电话号，传真号；

2，IP 地址范围；

3，DNS 服务器；

4，邮件服务器。

1.2 踩点相关搜索方法：

1，搜索网页。 site:xxx.com

2，链接搜索

目标网站所在的服务器可能有其他具有弱点的网站，可以进行迂回入侵，而且可以发现某些隐含的信息。

搜索方法介绍：通过各种搜索引擎：GOOGLE，http://www.dogpile.com， http://www.hotbot.com 等等

二，查点

2.1 确定目标的域名和相关的网络信息。

搜索方法；

Whois 查询，通过 Whois 数据库查询可以得到以下的信息：

1，注册机构：显示相关的注册信息和相关的 Whois 服务器；

2，机构本身：显示与某个特定机构相关的所有信息；

3，域名：显示与某个特定域名相关的所有信息

4，网络：显示与某个特定网络或单个 IP 地址相关的所有信息；

5，联系点:显示与某位特定人员相关的所有信息

搜索引擎站：http://www.infobear.com/whois.shtml

2.2 利用 ARIN 数据库可以查询某个域名所对应的网络地址分配信息。

相关搜索地址：http://ws.arin.net/cgi-bin/whois.pl

利用 http://whois.apnic.net/apnic-bin/whois2.pl 进行对 IP 地址的查询，以搜集有关的网络信息：

知道了目标所在的网络，可以进行迂回渗透，寻找薄弱点，进入目标网络，然后在 攻击目标。

2.3 DNS 信息查询

域名系统允许把一个 DNS 命名空间分割成多个区，各个去分别保存一个或多个 DNS 域 的名字信息。

区复制和区传送：

DNS 服务器之间是采用区传送的机制来同步和复制区内数据的。

区传送的安全问题不在于所传输的域名信息，而在于其配置是否正确。因为有些域 名信息当中包含了不应该公开的内部主机和服务器的域名信息。

相关工具：

1，Windows 下，可使用 nslookup,SamSpade;

2, UNIX 下：可使用 nslookup,dig,host,axfr

在 Windows 下的使用方法：

c:\>nslookup

Default server: 目标的 DNS 服务器

Address: 目标的 IP 地址

>set type=ANY //表示接受任何可能的 DNS 记录

>ls -d 163.com >zone.163.com.txt //获得目标域的相关记录,结果保存在 zone.163.com.txt

D，通过 Traceroute 获得网络的拓扑结构以及网络网络设备的地址。

相关工具;

Windows 下：Tracert 支持 ICMP 协议

UNIX 下：Traceroute 支持 ICMP 和 DNS 协议，由于多数防火墙已经过滤了 ICMP，所以 UNIX 下的 Traceroute 是不错的选择，而且使用-p n 选项可以自己指定使用的端口。

三，网络扫描

面对不同的网络，应该采用不用的扫描方法:

3.1 对于内部网络

可用类型很多，ICMP 协议是普遍要装上的，在内部网广播 ICMP 数 据包可以区分 WINDOWS 和 UNIX 系统,

发送类型为 8 的 ICMP 的 ECHO 请求,如果可以受到类型为 0 的 ECHO 回应,表明对方主机 是存活的。

相关工具介绍:

UNIX 下的：fping&gping

WINDOWS 下：Pinger 特点：速度快，多线程。

3.2 对于外部网络

可用类型也很多，涉及到的原理也有很多，例如：TCP 扫描，UDP 扫描等等。

其实我是很不愿意用扫描工具的，很容易使对方感觉到入侵事件的发生，不论是防火墙还是入侵检测系统都会或多或少的留下我们的脚印，如果遇到一个勤快的管理 员的话，那么这次入侵很可能以失败告终。

但使用与否依各个喜好而定了，有时候我们在测试网络或者主机的安全性时，就不能忽视他的存在了，首先，安全测试不是入侵，全面的测试对抵御黑客和蠕虫 的攻击是必要的，在这里推荐的端口扫描工具是 NMAP，因为他带有躲避 IDS 检测的机 制，重组了 TCP 的三次握手机制，慢扫描机制等等都是其他扫描工具无法比拟的，U DP 扫描是很不可靠的，原因有下几点：

这种扫描依靠 ICMP 端口不可达消息,如果发送端给目标一个感兴趣的端口发送了一个 UDP 数据包后，没有收到 ICMP 端口不可打消息，那么我们认为该端口处于打开状态。

该端口不可靠的原因：

1，路由器可能丢弃 UDP 分组；

2，很多的 UDP 服务不也不产生响应；

3，防火墙的常规配置是丢弃 UDP 分组（除 DNS 外）；

4，休眠状态的 UDP 端口是不会发送一个 ICMP 端口不可到达消息。

还有的扫描工具就是弱点扫描工具，这些工具综合各种漏洞信息构造漏洞数据库， 去探究存在漏洞没有打补丁的主机,当然也有针对特定漏洞的检测发现工具（脚本小 子能用，网络安全人员也弄用--双刃剑-：）

3.3 这里详细介绍对目标操作系统类型的检测原理：

3.3.1Telnet 标识和 TCP/IP 堆栈指纹：

1，网上许多的系统可以直接 Telnet 到目标，大多会返回欢迎信息的，返回的信息包 含了该端口所对应的服务软件的版本号，这个对于寻找这个版本的软件的漏洞很重 要，如果对方开了 Telnet，那么可以直接得到对方的系统类型和版本号，这个对于 挖掘系统的漏洞很重要（对于溢出来说，不同版本的系统和语言版本的系统来说， RET 地址，JMP ESP，地址是不同的）。

2，如今越来越多的管理员懂的了关闭功能标志，甚至提供伪造的欢迎信息。那么 TCP/IP 堆栈指纹是区分不同系统的好方法。

1，FIN 扫描

给打开的端口发送 FIN 包，RFC 793 规定不返回任何响应，例外的系统是：MS Wind ows,BSDI,CISCO,HP/UX,MVS 和 IRIX 都返回一个 RESET 包。

2，TCP 初始序列号（ISN）采样

这种方法利用了在实现 TCP 连接时使用不同的 ISN 模式识别系统，可以分成多种模式 ：传统的 64K 增加（旧 UNIX OS），随机增加（新版的 Solaris,IRIX,FreeBSD,Digital UNIX 和 Cray 等）， 真正随机（Linux 2.0.*,OpenVMS 和新版 AIX 等），Windows 系统使用所谓的“时间依 赖性”模型，即 ISN 的增加同某一个短固定的时间间隔有关系，有些主机始终使用固 定的 ISN，例如 3COM 集线器（使用 0x803)和 AppleLaserWriter 打印机（0xC7001）。

3，不分片位

目前许多系统在他们发送的包中使用 IP“不分片”位，这主要是想获得好的运行性 能，不过也不是所有的操作系统都有此功能，即使有，其实现的方式可能

也不同。 因此利用次位或许有利于我们收集更多的有关目标 OS 的信息。

4，TCP 初始窗

TCP 初始窗只是简单地测试返回包的窗口尺寸。Queso 和 Nmap 可以对实际的窗口进行 窗口跟踪。在很多操作系统中是一个常数。例如：AIX 是唯一使用 0x3F25 的操作系统 。对于完全重新编写代码的 NT 5 的 TCP 堆栈，使用 0x402E.

5,ACK 值

如果发送一个 FIN|PSH|URG，许多操作系统设置 ACK 等于初始序列号，而 Windows 和某 些打印机将发送 seq+1.如果发送一个 SYN|FIN|PSH|URG 到打开的端口，不同的 Windo ws 系统的实现将很不一致，有时返回 seq,有时返回 seq+1，甚至返回完全随机的数值 。

6，ICMP 错误消息机制

某些操作系统按照 RFC 1812 的建议，限制不同错误消息的发送速率。例如：Linux 内 核（在 net/ipv4/icmp.h 中定义）限制目标不可到达消息的产生速率为 4 秒种内 80 个 ，如果超过这个限制将有 1/4 的惩罚。测试方法是发送一大串包到某些随机选取的高 端口，然后计算返回的不可到达包的数目。

7，ICMP 消息引用（Message Quoting)

RFC 规定：ICMP 错误消息将引用一小部分导致错误消息包的 ICMP 消息内容。对于端口 不可达消息，几乎所有的实现都只发送所需要的 IP 头+8 字节。不过 Solaris 发送的内 容更多，而 Linux 发送的东西最多。这就是我们识别没有打开任何端口的 Linux 和 So laris 主机。

8，ICMP 错误消息回射完整性

主机对端口不可打错误消息将送回一小部分于是消息的内容。某些机器送回

的包中 包括的协议头部分已经被改变。例如，AIX 和 BSDI 送回的 IP 总长度是 20 字节。而系统 BSDI，FreeBSD,OpenBSD,ULTRIX 和 VAXen 则将原样送回你所发送的 IP 标识符。某些系 统 (AIX 和 FreeBSD 等) 将送回不一致或等于 0 的校验和。这同样适用于 UDP 校验和。 Nmap 对 ICMP 错误消息包进行九种不同的测试以标识系统之间的微笑差别。

9，TCP 选项

是实现 TCP/IP 协议时可选的一个部分功能，这跟不同的系统实现有关，这些选项都是挖掘可用信息的好方法。原因是：

1，他们都是可选项，不是所有主机都可以实现的；

2，如果你所发送的包中对某个选项进行了设置，只要目标支持，那么目标主机就返 回此选项；

3，可以在包中设置所有的选项进行测试。

例如：Nmap 在每个探测包中设置所有的选项来进行测试：

Windows Scale=10;NOP;Max Segment Size=265;Timestamp;End of Ops;

从返回的的包中查看这些选项，就知道了什么系统支持他们。

还有一种被动操作系统识别方法，就是监控不同系统之间网络包的情况来判断目标 的操作系统类型，siphon 被用来进行这方面的测试，这个工作原理如下：

签名：

主要 TCP 的四个字段判断：

1，TTL：出站的包的存活时间；

2，Window size：窗口大小；

3，DF：是否设置了不准分片位；

4，TOS：是否设置了服务类型。

综合这些信息可以大概判断出目标的系统，但也不能 100%确认。

四，查点

利用查点技术可以得到比前面讲的更多更具体的有用信息，例如：帐户信息等。

4.1 Windows 系统查点技术

利用 NetBIOS 规则，首先介绍 NetBIOS,NetBOIS 位于 TCP/IP 之上，定义了多个 TCP 和 U DP 端口。

----TCP

（1），139：nbsession:NetBOIS 会话。

例如：net use \\IP\ipc$ " " /user:" ".

（2），42：WINS：Windows Internet 名字系统（UDP 端口也是 42）。

----UDP

（1）137：nbname:名字查询。

例如：nbtstat -A IP //03 中显示的不是计算机名就是用户名

（2）138：nbdatagram:UDP 数据报服务

例如：net send /d:domain-name "Hello"

得到用户名利用到了 IPC$空会话和 sid 工具。sid 工具由两个小工具组成：user2sid 和 sid2user.user2sid 获得用户名或组名的 sid;sid2user 则是输入一个 sid 而获得相 应用户名的和组名，sid 就是在创建用户时而创建的，相当于 UNIX 系统下的 UID,WIN 系统权限的检查就是通过对 SID 的检查的。一个 sid 是由一长串数字组成的，其中包 括两个部分，前一部分用来唯一标识一个域，后一部分唯一标识一个用户名，这部 分数字被称作 rid，既相对标识符，rid 有一定的规律，

其取值总是从 500 开始的，超 级管理员的 rid 总是 500，而 GUEST 用户的 rid 总是 501；而新建立的帐户的 rid 从 1000 开始。

具体的步骤：

c:\net use \\IP\ipc$ " " /user:" "

c:\user2sid \\IP guest //得到了 SID 的前半部分

s-1-5-21-1123561945-1580818891-1957994488-501

s 是 sid 的前缀,后面跟的是 1 表示版本号,5 用于标识发放 sid 的授权实体，5 指 NT/2 000。21-1123561945-1580818891-1957994488 唯一地标识域和工作组。不同的用户 只是最后的相对标识符不一样。现在用 sid2user 查询系统的用户名了：

c:\sid2user \\IP 5 21 1123561945 1580818891 1957994488 500

name is cookie

domain is condor

c:\sid2user \\IP 5 21 1123561945 1580818891 1957994488 1001

SNMP 查点：通过默认的管理群字符串 PUBLIC 读取特性，可以得到系统的一些信息， 具体有：接口表，路由表及 ARP 表，TCP 表和 UDP 表，设备表和存储表，进程表和软件 表，用户表，共享表。

SNMP 工具，snmputil.exe

例如：

1，或者网络接口数目：

c:\snmputil get localhost public .1.3.6.1.2.1.2.1.0

2,显示所有的 SNMP 变量内容

c:\snmputil walk localhost public .1.3

4.2 UNIX 类系统的查点技术

1，$showmount -e www.target.com　//前提 2049 号端口开着（NFS）

2，$finger @www.target.com //还有 rusers

3, $telnet www.target.com 25

vrfy root　//证实是否有 root

expn adm

quit

五，具体的分析漏洞

针对特定目标进行了以上分析后，总结出最好的入侵思路，选择入侵工具，做好入侵的准备工作是必须，有时入侵时间的选择也是很重要的，因为会涉及到正常的公司网络的正常通信，甚至会使恶意的网络在你入侵测试就发生了，最直接的漏洞利 用方法，我认为是溢出漏洞了，因为他直接就可以得到对方的系统权限，返回一个和在本地一样的 SHELL 环境，此时无所不能：

溢出攻击的分类有：

5.1 WINDOWS 下的和 UNIN 下的攻击分类

一般原理，就用户提交的参数范围超过了在内存中保存的本地变量的范围，而程序 或者系统并没有对输入的参数进行合理的长度检查，导致了被调用函数的返回地址 被覆盖，如果用一个跳转到我们提交的 shellcode 的地方的地址代替，那么我们的 s hellcode 就可以运行，成功得到了目标的系统权限。

此外还有格式化串漏洞，导致这个漏洞的原因是在处理用户数据的参数时没有过滤 用户提交的，格式化符号，例如%n 这个将允许输出的参数的个数保存在内存中，恶 意构造此漏洞用户将会向内存的任何位置写 SHELLCODE 的地址。

5.2 常见漏洞类型

UNIX 下的本地漏洞很多，挖掘起来也较容易，他主要有以下几种类型：

5.2.1 环境欺骗

一般指 PATH 环境变量的欺骗，就是说如果一个特权的程序执行了一个外部的命令， 那么我们可以简单的构造这个外部命令程序，然后修改 PATH 使这个特权程序能够去 首先执行我们构造的外部命令程序，而这个外部的命令程序是一个去得 SHELL 的程序

5.2.2 竞争条件

竞争条件一般指时序竞争，例如：

```
fp=fopen("test.log","w+");

chown("test.log",getuid(),getgid());
```

原理也很简单，就是如果当前的程序运行时权限是 euid=root,uid=当前用户，由于文件 test.log 在打开会执行将文件的属主改为当前用户，所以我们可以在执行完 fo pen 之后，chown 之前删了 test.log，而创建了一个到/etc/passwd 的符号链接，这样 就会将/etc/passwd 文件的属主改为当前的用户，当前的用户就可以在 passwd 文件中 将自己的 uid 改为 0，这样就取得了 system 权限。

5.3.3 溢出和格式串漏洞

导致这些漏洞的数据来源主要是：

1，命令行参数

2，环境变量

3，特定格式文件的读取

4，用户交互时的输入

缓冲溢出的漏洞是有以下一些函数引起的：

1. strcpy
2. strcat
3. sprintf
4. vsprintf

格式化串的漏洞和以下一些函数有关：

1. print/vprintf
2. fprintf/vfprintf
3. sprintf/vsprintf
4. snprintf/vsnprintf

利用工具有 objdump,elfedump 查看目标是否有不安全的以上不安全的函数，如果有 可以进行黑盒测试，进而进行返汇编分析程序的上下文和执行流程，利用 strings 可 以静态查找目标的环境变量。

六，攻击 WWW

现在的入侵事件，攻击 WWW 居多，原因也很简单，那就是程序员在编写 WEB 脚本程序 时更本不注重安全因素，导致了上传 shell,提升权限之类的严重后果，入侵渗透测试主要通过以下几个方面进行测试：

1，搜索 SQL 注入点；

2，搜索特定目录和文件，例如：上传程序文件，这个利用价值也很大；

3，寻找管理员登陆网页，进行字典或者 SQL 饶过入侵；

4，寻找 WEB 程序的源代码，进行漏洞挖掘，主要涉及的漏洞类型有：SQL 注入，文件 包含漏洞，目录跳转漏洞，以脚本文件格式保存错误日志漏洞，上传漏洞；

5，在代码审核时，不要忘记对程序员犯的逻辑错误进行查看，例如：函数书写错误

6，总是，漏洞的成因归根到底是由于对用户的输入没有进行严格的过滤。

七，其他的入侵

1，针对数据库 MSSQL，MYSQL，ORACLE 等数据库的入侵；

2，针对路由，防火墙，IDS 等网络设备的渗透

3，无线入侵渗透

八，入侵渗透成功以后

1，在成功得到系统级别的权限以后，就要在目标留下后门方便以后进入，当然清楚 日志是最为重要的收尾工作，这些方面也有很多的技术可以讨论，例如：后门的隐藏（WIN 下的 ADS 是一个不错的隐藏程序的东西），日志的有选择删除及其伪造等等。