

内存取证

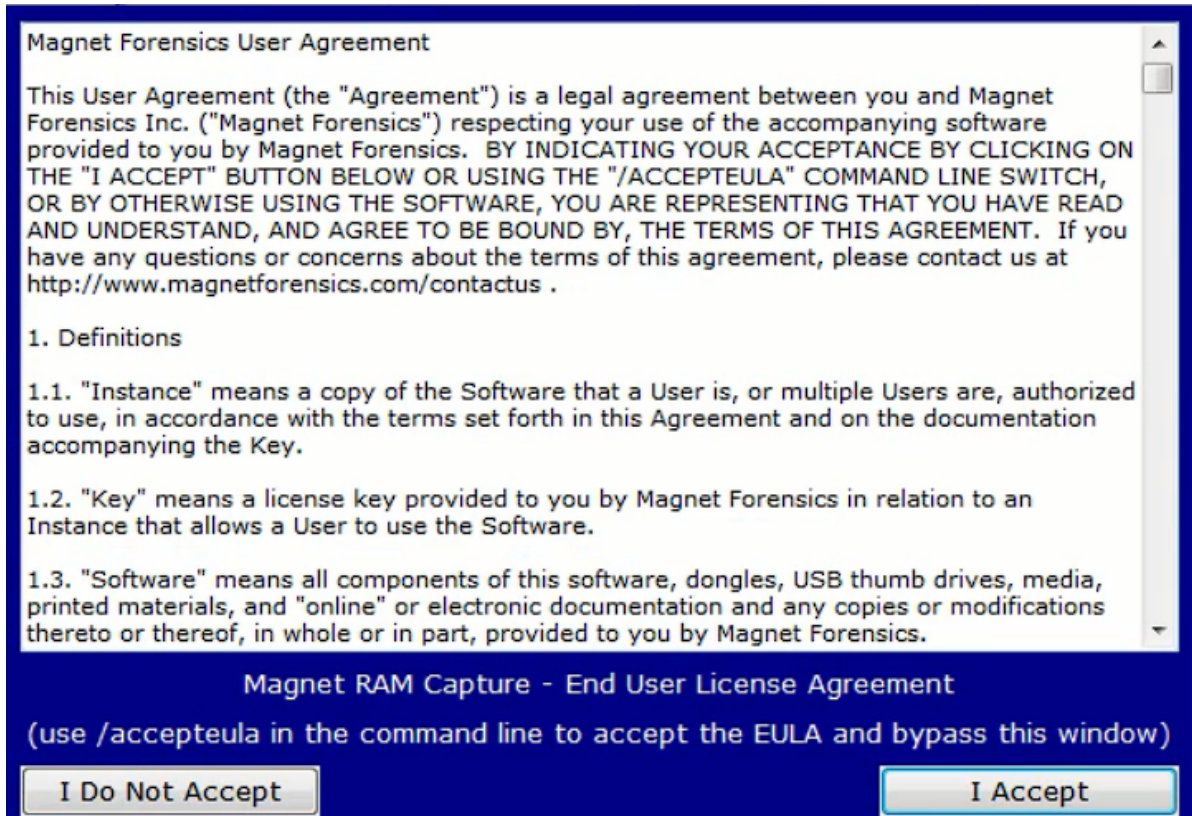
一、获取windows系统内存镜像

1、MAGNET RAM Capture

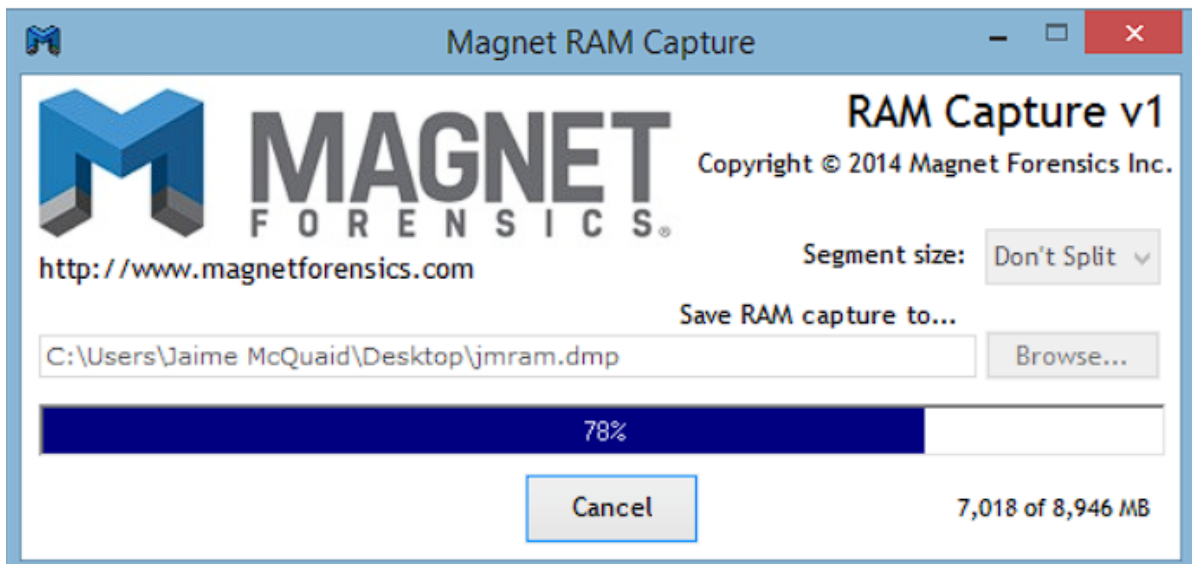
MAGNET RAM Capture是由MAGNET公司开发的一款免费制作内存镜像工具，体积小还可以对内存镜像设置分段。

使用步骤：

- ① 打开软件，在用户协议界面点击“I Accept”。



- ② 点击“Browse”，选择制作镜像保存的目录，并自定义文件名，后单击“start”，开始制作镜像。



2、DumpIt

DumpIt 是一款绿色免安装的 windows 内存镜像取证工具。利用它我们可以轻松地将一个系统的完整内存镜像下来。

① 直接双击运行 DumpIt.exe 文件

 DumpIt.exe	2014/8/17 20:24	应用程序	203 KB
 README.txt	2014/8/17 20:24	文本文档	1 KB

② 输入y并回车，提取的镜像被保存在当前目录下，保存格式为 raw。

```
C:\Users\Administrator\Downloads\All-In-USB-master\All-In-USB-master\utilities\DumpIt\DumpIt.exe
DumpIt - v1.3.2.20110401 - One click memory memory dumper
Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

Address space size:      10183770112 bytes ( 9712 Mb)
Free space size:        29278060544 bytes ( 27921 Mb)

* Destination = \\?\C:\Users\Administrator\Downloads\All-In-USB-master\All-In-USB-master\utilities\DumpIt\DESKTOP-TH
DVUC-20210726-105420.raw

--> Are you sure you want to continue? [y/n] y
+ Processing...
```

二、取证工具——volatility使用方式

Volatility是一款非常强大的内存取证工具,它是来自全世界的数百位知名安全专家合作开发的一套工具,可以用于windows,linux,mac osx,android等系统内存取证。Volatility是一款开源内存取证框架,能够对导出的内存镜像进行分析,通过获取内核数据结构,使用插件获取内存的详细情况以及系统的运行状态。

官网下载地址: [Volatility 2.6 Release \(volatilityfoundation.org\)](http://volatilityfoundation.org)

Volatility 2.6 (Windows 10 / Server 2016)

This release improves support for Windows 10 and adds support for Windows Server 2016, Mac OS Sierra 10.12, and Linux with KASLR kernels. A lot of bug fixes went into this release as well as performance enhancements (especially related to page table parsing and virtual address space scanning). See below for a more detailed list of the changes in this version.

This release also coincides with the [Community repo](#) - a collection of Volatility plugins written and maintained by authors in the forensics community. Many of these are the result of the last 4 years of [Volatility plugin contests](#), but some were just written for fun. Either way, its an entire arsenal of plugins that you can easily extend into your existing Volatility installation.

Released: December 2016

- [Download the Volatility 2.6 Windows Standalone Executable \(x64\)](#)
- [Download the Volatility 2.6 Mac OS X Standalone Executables \(x64\)](#)
- [Download the Volatility 2.6 Linux Standalone Executables \(x64\)](#)
- [Download the Volatility 2.6 Source Code \(.zip\)](#)
- [Download the Integrity Hashes](#)
- [View the README](#)
- [View the CREDITS](#)

Windows平台

Linux平台

(一) .使用方法

此处以windows版的程序【volatility_2.6_win64_standalone.exe】为例，若使用源码包进行使用，更改程序名字即可。

1.查看帮助

```
1 | volatility_2.6_win64_standalone.exe --h
```

具体可查看本手册最后【附件10-1】内容

2.查看插件

```
1 | volatility_2.6_win64_standalone.exe --info
```

具体可查看本手册最后【附件10-2】内容

3.分析镜像系统信息

```
1 | # imageinfo
2 | volatility_2.6_win64_standalone.exe -f xxx.raw imageinfo
```

此步骤很重要，后面所有操作都要基于此步骤分析出的系统信息，添加参数操作

4.查看进程

```
1 | # pslist
2 | volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX pslist
```

5.进程转储

```
1 | # memdump 导出进程数据
2 | # -p 导出的进程PID值
3 | # -D 导出路径
4 | volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX memdump -p 2012
   | -D ./
```

6.查看用户名

```
1 | # printkey
2 | volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX printkey -k
   | "SAM\Domains\Account\Users\Names"
```

7.查看用户密码

```
1 # hashdump
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx hashdump
```

8.查看主机名

```
1 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx printkey -k
   "ControlSet001\Control\ComputerName\ComputerName"
```

9.查看cmd进程

```
1 # cmdscan
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx cmdscan
```

10.扫描文件

```
1 # filescan
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx filescan
```

11.导出文件

```
1 # dumpfiles
2 # -Q 文件的偏移
3 # -D 保存路径
4 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx dumpfiles -Q
   0x00000123456 -D ./
```

12.查看进程树

```
1 # pstree
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=xxx pstree
```

13.查看某个进程的DLL

```
1 # dlllist
2 # -p 端口号
3 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX dlllist -p 1234
```

14.查看剪贴板

```
1 # clipboard
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX clipboard
```

15.查看ie浏览器历史记录

```
1 # iehistory
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX iehistory
```

16.查看环境变量

```
1 # envvars
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX envvars
```

17.查看程序版本信息

```
1 # verinfo
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX verinfo
```

18.查看开机自启的程序

```
1 # shimcache
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX shimcache
```

19.查看系统网络连接情况

```
1 # netscan
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX netscan
```

20.查看注册表配置单元

```
1 # hivelist
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX hivelist
```

21.查看系统服务

```
1 # svcsan
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX svcsan
```

22.导出进程的程序

```
1 # svcsan
2 volatility_2.6_win64_standalone.exe -f xxx.raw --profile=XXX procdump -p
   [PID] -D ./
```

23 导出进程内容

```
1 # memdump
2 .\volatility_2.6_win64_standalone.exe -f .\OtterCTF.vmem --profile=win7SP1x64
   memdump -p [PID] -D ./
```

附件10-1

```
1 Volatility Foundation Volatility Framework 2.6
2 用法: volatility - 内存取证分析平台
3
4 Options:
5   -h, --help           列出所有可用选项及其默认值
6                       默认值可以在配置文件中设置
7                       (/etc/volatilityrc)
8   --conf-file=/home/kali/.volatilityrc
9                       基于用户的配置文件
10  -d, --debug           调试Volatility
11  --plugins=PLUGINS    要使用的其他插件目录（冒号分隔）
12  --info               打印所有注册对象的信息
13  --cache-directory=/home/kali/.cache/volatility
14                       存放缓存文件的目录
15  --cache              使用缓存
16  --tz=TZ              设置 (Olson) 时区以使用 pytz（如果已安装）或 tzset 显示时
   间戳
```

```

17 -f FILENAME, --filename=FILENAME
18         打开图像时使用的文件名
19 --profile=winXPSP2x86
20         要加载的配置文件的名称 (使用 --info 查看支持的配置文件列表)
21 -l LOCATION, --location=LOCATION
22         从中加载地址空间的 URN 位置
23 -w, --write
24         启用写支持
25 --dtb=DTB
26         DTB 地址
27 --shift=SHIFT
28         Mac KASLR 移位地址
29 --output=text
30         以这种格式输出 (支持特定于模块, 请参阅下面的模块输出选项)
31 --output-file=OUTPUT_FILE
32         在此文件中写入输出
33 -v, --verbose
34         详细信息
35 -g KDBG, --kdbg=KDBG 指定一个 KDBG 虚拟地址 (注意: 对于 64 位 windows 8 及更高
36 版本, 这是 KdCopyDataBlock 的地址)
37 --force
38         强制使用可疑配置文件
39 -k KPCR, --kpcr=KPCR 指定特定的 KPCR 地址
40 --cookie=COOKIE
41         指定 nt!ObHeaderCookie 的地址 (仅适用于 windows 10)
42
43 支持的插件命令:
44
45     amcache        查看AmCache应用程序痕迹信息
46     apihooks       检测内核及进程的内存空间中的API hook
47     atoms          列出会话及窗口站atom表
48     atomscan       Atom表的池扫描(Pool scanner)
49     auditpol       列出注册表HKLMSECURITYPolicyPolAdtEv的审计策略信息
50     bigpools       使用BigPagePoolScanner转储大分页池(big page pools)
51     bioskbd        从实时模式内存中读取键盘缓冲数据(早期电脑可以读取BIOS开机
52 密码)
53     cachedump      获取内存中缓存的域帐号的密码哈希
54     callbacks      打印全系统通知例程
55     clipboard      提取windows剪贴板中的内容
56     cmdline        显示进程命令行参数
57     cmdscan        提取执行的命令行历史记录 (扫描_COMMAND_HISTORY信息)
58     connections    打印系统打开的网络连接(仅支持windows XP 和2003)
59     connscan       打印TCP连接信息
60     consoles       提取执行的命令行历史记录 (扫描_CONSOLE_INFORMATION信息)
61     crashinfo      提取崩溃转储信息
62     deskscan       tagDESKTOP池扫描(Poolscanner)
63     devicetree     显示设备树信息
64     dlldump        从进程地址空间转储动态链接库
65     dlllist        打印每个进程加载的动态链接库列表
66     driverirp      IRP hook驱动检测
67     drivermodule   关联驱动对象至内核模块
68     driverscan     驱动对象池扫描
69     dumpcerts      提取RAS私钥及SSL公钥
70     dumpfiles      提取内存中映射或缓存的文件
71     dumpregistry   转储内存中注册表信息至磁盘
72     editbox        查看Edit编辑控件信息 (ListBox正在实验中)
73     envvars        显示进程的环境变量
74     eventhooks     打印windows事件hook详细信息
75     evtlogs        提取windows事件日志 (仅支持XP/2003)
76     filescan       提取文件对象 (file objects) 池信息
77     gahti          转储用户句柄 (handle) 类型信息
78     gditimers      打印已安装的GDI计时器(timers)及回调(callbacks)

```

70	gdt	显示全局描述符表(Global Descriptor Table)
71	getservicesids	获取注册表中的服务名称并返回SID信息
72	getsids	打印每个进程的SID信息
73	handles	打印每个进程打开的句柄的列表
74	hashdump	转储内存中的Windows帐户密码哈希(LM/NTLM)
75	hibinfo	转储休眠文件信息
76	hivedump	打印注册表配置单元信息
77	hivelist	打印注册表配置单元列表
78	hivescan	注册表配置单元池扫描
79	hpakextract	从HPAK文件(Fast Dump格式)提取物理内存数据
80	hpakinfo	查看HPAK文件属性及相关信息
81	idt	显示中断描述符表(Interrupt Descriptor Table)
82	iehistory	重建IE缓存及访问历史记录
83	imagecopy	将物理地址空间导出原生DD镜像文件
84	imageinfo	查看/识别镜像信息
85	impscan	扫描对导入函数的调用
86	joblinks	打印进程任务链接信息
87	kdbgscan	搜索和转储潜在KDBG值
88	kpcrscan	搜索和转储潜在KPCR值
89	ldrmodules	检测未链接的动态链接DLL
90	lsadump	从注册表中提取LSA密钥信息(已解密)
91	machinfo	转储Mach-O文件格式信息
92	malfind	查找隐藏的和插入的代码
93	mbrparser	扫描并解析潜在的主引导记录(MBR)
94	memdump	转储进程的可寻址内存
95	memmap	打印内存映射
96	messagehooks	桌面和窗口消息钩子的线程列表
97	mftparser	扫描并解析潜在的MFT条目
98	moddump	转储内核驱动程序到可执行文件的示例
99	modscan	内核模块池扫描
100	modules	打印加载模块的列表
101	multiscan	批量扫描各种对象
102	mutantscan	对互斥对象池扫描
103	notepad	查看记事本当前显示的文本
104	objtypescan	扫描窗口对象类型对象
105	patcher	基于页面扫描的补丁程序内存
106	poolpeek	可配置的池扫描器插件
107	printkey	打印注册表项及其子项和值
108	privs	显示进程权限
109	procdump	进程转储到一个可执行文件示例
110	pslist	按照EPROCESS列表打印所有正在运行的进程
111	psscan	进程对象池扫描
112	pstree	以树型方式打印进程列表
113	psxview	查找带有隐藏进程的所有进程列表
114	qemuinfo	转储Qemu信息
115	raw2dmp	将物理内存原生数据转换为windbg崩溃转储格式
116	screenshot	基于GDI Windows的虚拟屏幕截图保存
117	servicediff	Windows服务列表(ala Plugx)
118	sessions	_MM_SESSION_SPACE的详细信息列表(用户登录会话)
119	shellbags	打印Shellbags信息
120	shimcache	解析应用程序兼容性Shim缓存注册表项
121	shutdowntime	从内存中的注册表信息获取机器关机时间
122	sockets	打印已打开套接字列表
123	sockscan	TCP套接字对象池扫描
124	ssdt	显示SSDT条目

125	strings	物理到虚拟地址的偏移匹配(需要一些时间, 带详细信息)
126	svcsan	windows服务列表扫描
127	symlinkscan	符号链接对象池扫描
128	thrdscan	线程对象池扫描
129	threads	调查_ETHREAD 和_KTHREADS
130	timeliner	创建内存中的各种痕迹信息的时间线
131	timers	打印内核计时器及关联模块的DPC
132	truecryptmaster Recover	恢复TrueCrypt 7.1a主密钥
133	truecryptpassphrase	查找并提取TrueCrypt密码
134	truecryptsummary	TrueCrypt摘要信息
135	unloadedmodules	打印卸载的模块信息列表
136	userassist	打印注册表中UserAssist相关信息
137	userhandles	转储用户句柄表
138	vaddump	转储VAD数据为文件
139	vadinfo	转储VAD信息
140	vadtrees	以树形方式显示VAD树信息
141	vadwalk	显示遍历VAD树
142	vboxinfo	转储Virtualbox信息(虚拟机)
143	verinfo	打印PE镜像中的版本信息
144	vmwareinfo	转储VMware VMSS/VMSN 信息
145	volshell	内存镜像中的shell
146	windows	打印桌面窗口(详细信息)
147	wintree	Z顺序打印桌面窗口树
148	wndscan	池扫描窗口站
149	yarascan	以Yara签名扫描进程或内核内存

附件10-2

```

1 Volatility Foundation Volatility Framework 2.6
2
3 Profiles
4 -----
5 VistaSP0x64 - windows Vista SP0 x64 的配置文件
6 VistaSP0x86 - windows Vista SP0 x86 的配置文件
7 VistaSP1x64 - windows Vista SP1 x64 的配置文件
8 VistaSP1x86 - windows Vista SP1 x86 的配置文件
9 VistaSP2x64 - windows Vista SP1 x86 的配置文件
10 VistaSP2x86 - windows Vista SP2 x64 的配置文件
11 win10x64 - windows 10 x64 的配置文件
12 win10x64_10586 - windows 10 x64 的配置文件 (10.0.10586.306 / 2016-04-23)
13 win10x64_14393 - windows 10 x64 的配置文件 (10.0.14393.0 / 2016-07-16)
14 win10x86 - windows 10 x86 的配置文件
15 win10x86_10586 - windows 10 x86 的配置文件 (10.0.10586.420 / 2016-05-28)
16 win10x86_14393 - windows 10 x86 的配置文件 (10.0.14393.0 / 2016-07-16)
17 win2003SP0x86 - windows 2003 SP0 x86 的配置文件
18 win2003SP1x64 - windows 2003 SP0 x86 的配置文件
19 win2003SP1x86 - windows 2003 SP1 x86 的配置文件
20 win2003SP2x64 - windows 2003 SP1 x86 的配置文件
21 win2003SP2x86 - windows 2003 SP2 x86 的配置文件
22 win2008R2SP0x64 - windows 2008 R2 SP0 x64 的配置文件

```

23	Win2008R2SP1x64	- windows 2008 R2 SP1 x64 的配置文件
24	Win2008R2SP1x64_23418	- windows 2008 R2 SP1 x64 的配置文件 (6.1.7601.23418 / 2016-04-09)
25	Win2008SP1x64	- windows 2008 SP1 x64 的配置文件
26	Win2008SP1x86	- windows 2008 SP1 x86 的配置文件
27	Win2008SP2x64	- windows 2008 SP2 x64 的配置文件
28	Win2008SP2x86	- windows 2008 SP2 x86 的配置文件
29	Win2012R2x64	- windows Server 2012 R2 x64 的配置文件
30	Win2012R2x64_18340	- windows Server 2012 R2 x64 的配置文件 (6.3.9600.18340 / 2016-05-13)
31	Win2012x64	- windows Server 2012 x64 的配置文件
32	Win2016x64_14393	- windows Server 2016 x64 的配置文件 (10.0.14393.0 / 2016-07-16)
33	Win7SP0x64	- windows 7 SP0 x64 的配置文件
34	Win7SP0x86	- windows 7 SP0 x86 的配置文件
35	Win7SP1x64	- windows 7 SP1 x64 的配置文件
36	Win7SP1x64_23418	- windows 7 SP1 x64 的配置文件 (6.1.7601.23418 / 2016-04-09)
37	Win7SP1x86	- windows 7 SP1 x86 的配置文件
38	Win7SP1x86_23418	- windows 7 SP1 x86 的配置文件 (6.1.7601.23418 / 2016-04-09)
39	Win81U1x64	- windows 8.1 更新 1 x64 的配置文件
40	Win81U1x86	- windows 8.1 更新 1 x86 的配置文件
41	Win8SP0x64	- windows 8 x64 的配置文件
42	Win8SP0x86	- windows 8 x86 的配置文件
43	Win8SP1x64	- windows 8.1 x64 的配置文件
44	Win8SP1x64_18340	- windows 8.1 x64 的配置文件 (6.3.9600.18340 / 2016-05-13)
45	Win8SP1x86	- windows 8.1 x86 的配置文件
46	WinXPSP1x64	- windows XP SP1 x64 的配置文件
47	WinXPSP2x64	- windows XP SP2 x64 的配置文件
48	WinXPSP2x86	- windows XP SP2 x86 的配置文件
49	WinXPSP3x86	- windows XP SP3 x86 的配置文件
50		
51		
52	Address Spaces	
53	-----	
54	AMD64PagedMemory	- 标准 AMD 64 位地址空间
55	ArmAddressSpace	- ARM 处理器的地址空间
56	FileAddressSpace	- 这是一个直接文件 AS.
57	HPAKAddressSpace	- 此 AS 支持 HPAK 格式
58	IA32PagedMemory	- 标准 IA-32 分页地址空间
59	IA32PagedMemoryPae	- 此类实现 IA-32 PAE 分页地址空间
60	LimeAddressSpace	- Lime 的地址空间
61	LinuxAMD64PagedMemory	- Linux 特定的 AMD 64 位地址空间
62	Mach0AddressSpace	- mach-o 文件的地址空间以支持 atc-ny 内存读取器
63	OSXPmemELF	- 这个 AS 支持 virtualBox ELF64 coredump 格式
64	QemuCoreDumpElf	- 这个 AS 支持 Qemu ELF32 和 ELF64 核心转储格式
65	VMwareAddressSpace	- 此 AS 支持 VMware 快照 (VMSS) 和保存状态 (VMSS) 文件
66	VMwareMetaAddressSpace	- 此 AS 支持带有 VMSN/VMSS 元数据的 VMEM 格式
67	VirtualBoxCoreDumpElf64	- 这个 AS 支持 virtualBox ELF64 coredump 格式
68	Win10AMD64PagedMemory	- windows 10 特定的 AMD 64 位地址空间
69	WindowsAMD64PagedMemory	- windows 特定的 AMD 64 位地址空间
70	WindowsCrashDumpSpace32	- 这个 AS 支持 windows 崩溃转储格式

71	WindowsCrashDumpSpace64	- 此 AS 支持 windows Crash Dump 格式
72	WindowsCrashDumpSpace64BitMap	- 此 AS 支持 windows BitMap Crash Dump 格式
73	WindowsHiberFileSpace32	- 这是 windows 休眠文件的休眠地址空间
74		
75		
76	Plugins	
77	-----	
78	amcache	- 打印 AmCache 信息
79	apihooks	- 检测进程和内核内存中的 API 挂钩
80	atoms	- 打印会话和窗口站原子表
81	atomscan	- 原子表的池扫描器
82	auditpol	- 从 HKLM\SECURITY\Policy\PolAdtEv 打印出审计策略
83	bigpools	- 使用 BigPagePoolScanner 转储大页面池
84	bioskbd	- 从实模式内存中读取键盘缓冲区
85	cachedump	- 从内存中转储缓存的域哈希
86	callbacks	- 打印系统范围的通知例程
87	clipboard	- 提取 windows 剪贴板的内容
88	cmdline	- 显示进程命令行参数
89	cmdscan	- 通过扫描 _COMMAND_HISTORY 来提取命令历史记录
90	connections	- 打印打开的连接列表 [仅限 windows XP 和 2003]
91	connscan	- 用于 tcp 连接的池扫描器
92	consoles	- 通过扫描 _CONSOLE_INFORMATION 提取命令历史记录
93	crashinfo	- 转储崩溃转储信息
94	deskscan	- tagDESKTOP (台式机) 的 Poolscanner
95	devicetree	- 显示设备树
96	dlldump	- 从进程地址空间转储 DLL
97	dlllist	- 打印每个进程加载的 dll 列表
98	driverirp	- 驱动程序 IRP 挂钩检测
99	drivermodule	- 将驱动程序对象关联到内核模块
100	driverscan	- 驱动程序对象的池扫描器
101	dumpcerts	- 转储 RSA 私有和公共 SSL 密钥
102	dumpfiles	- 提取内存映射和缓存文件
103	dumpregistry	- 将注册表文件转储到磁盘
104	editbox	- 显示有关编辑控件的信息 (列表框实验)
105	envvars	- 显示进程环境变量
106	eventhooks	- 在 windows 事件挂钩上打印详细信息
107	evtlogs	- 提取 windows 事件日志 (仅限 XP/2003)
108	filescan	- 文件对象的池扫描器
109	gahti	- 转储 USER 句柄类型信息
110	gditimers	- 打印已安装的 GDI 计时器和回调
111	gdt	- 显示全局描述符表
112	getservicesids	- 获取 Registry 中的服务名称并返回计算的 SID
113	getsids	- 打印拥有每个进程的 SID
114	handles	- 打印每个进程的打开句柄列表
115	hashdump	- 从内存中转储密码哈希 (LM/NTLM)
116	hibinfo	- 转储休眠文件信息
117	hivedump	- 打印注册表
118	hivelist	- 打印注册表配置单元列表
119	hivescan	- 注册表配置单元的池扫描程序
120	hpakextract	- 从 HPAK 文件中提取物理内存
121	hpakinfo	- 有关 HPAK 文件的信息
122	idt	- 显示中断描述符表
123	iehistory	- 重建 Internet Explorer 缓存/历史
124	imagecopy	- 将物理地址空间复制为原始 DD 映像
125	imageinfo	- 识别图像的信息

126	<code>impscan</code>	- 扫描对导入函数的调用
127	<code>joblinks</code>	- 打印进程作业链接信息
128	<code>kdbgsan</code>	- 搜索和转储潜在的 <code>KDBG</code> 值
129	<code>kpcrscan</code>	- 搜索和转储潜在的 <code>KPCR</code> 值
130	<code>ldrmodules</code>	- 检测未链接的 <code>DLL</code>
131	<code>limeinfo</code>	- 转储 <code>Lime</code> 文件格式信息
132	<code>linux_apihooks</code>	- 检查用户态 <code>apihooks</code>
133	<code>linux_arp</code>	- 打印 <code>ARP</code> 表
134	<code>linux_aslr_shift</code>	- 自动检测 <code>Linux ASLR shift</code>
135	<code>linux_banner</code>	- 打印 <code>Linux</code> 横幅信息
136	<code>linux_bash</code>	- 从 <code>bash</code> 进程内存中恢复 <code>bash</code> 历史记录
137	<code>linux_bash_env</code>	- 恢复进程的动态环境变量
138	<code>linux_bash_hash</code>	- 从 <code>bash</code> 进程内存中恢复 <code>bash</code> 哈希表
139	<code>linux_check_afinfo</code>	- 验证网络协议的操作函数指针
140	<code>linux_check_creds</code>	- 检查是否有进程共享凭证结构
141	<code>linux_check_evt_arm</code>	- 检查异常向量表以查找系统调用表挂钩
142	<code>linux_check_fop</code>	- 检查 <code>rootkit</code> 修改的文件操作结构
143	<code>linux_check_idt</code>	- 检查 <code>IDT</code> 是否已被更改
144	<code>linux_check_inline_kernel</code>	- 检查内联内核挂钩
145	<code>linux_check_modules</code>	- 将模块列表与 <code>sysfs</code> 信息进行比较（如果可用）
146	<code>linux_check_syscall</code>	- 检查系统调用表是否已更改
147	<code>linux_check_syscall_arm</code>	- 检查系统调用表是否已更改
148	<code>linux_check_tty</code>	- 检查 <code>tty</code> 设备的钩子
149	<code>linux_cpuinfo</code>	- 打印每个活动处理器的信息
150	<code>linux_dentry_cache</code>	- 从 <code>dentry</code> 缓存中收集文件
151	<code>linux_dmesg</code>	- 收集 <code>dmesg</code> 缓冲区
152	<code>linux_dump_map</code>	- 将选定的内存映射写入磁盘
153	<code>linux_dynamic_env</code>	- 恢复进程的动态环境变量
154	<code>linux_elfs</code>	- 在进程映射中查找 <code>ELF</code> 二进制文件
155	<code>linux_enumerate_files</code>	- 列出文件系统缓存引用的文件
156	<code>linux_find_file</code>	- 列出并从内存中恢复文件
157	<code>linux_getcwd</code>	- 列出每个进程的当前工作目录
158	<code>linux_hidden_modules</code>	- 雕刻内存以查找隐藏的内核模块
159	<code>linux_ifconfig</code>	- 收集活动接口
160	<code>linux_info_regs</code>	- 就像 <code>GDB</code> 中的“信息寄存器”。它打印出所有
161	<code>linux_iomem</code>	- 提供类似于 <code>/proc/iomem</code> 的输出
162	<code>linux_kernel_opened_files</code>	- 列出从内核中打开的文件
163	<code>linux_keyboard_notifiers</code>	- 解析键盘通知器调用链
164	<code>linux_ldrmodules</code>	- 将 <code>proc</code> 映射的输出与 <code>libdl</code> 中的库列表进行比较
165	<code>linux_library_list</code>	- 列出加载到进程中的库
166	<code>linux_librarydump</code>	- 将进程内存中的共享库转储到磁盘
167	<code>linux_list_raw</code>	- 列出具有混杂套接字的应用程序
168	<code>linux_lsmod</code>	- 收集加载的内核模块
169	<code>linux_lsof</code>	- 列出文件描述符及其路径
170	<code>linux_malfind</code>	- 寻找可疑的进程映射
171	<code>linux_memap</code>	- 转储 <code>linux</code> 任务的内存映射
172	<code>linux_moddump</code>	- 提取加载的内核模块
173	<code>linux_mount</code>	- 收集挂载的 <code>fs/devices</code>
174	<code>linux_mount_cache</code>	- 从 <code>kmem_cache</code> 收集挂载的 <code>fs/devices</code>
175	<code>linux_netfilter</code>	- 列出 <code>Netfilter</code> 钩子
176	<code>linux_netscan</code>	- 雕刻网络连接结构
177	<code>linux_netstat</code>	- 列出打开的套接字
178	<code>linux_pidhashtable</code>	- 通过 <code>PID</code> 哈希表枚举进程
179	<code>linux_pkt_queues</code>	- 将每个进程的数据包队列写入磁盘
180	<code>linux_plthook</code>	- 扫描 <code>ELF</code> 二进制文件的 <code>PLT</code> 以获取非需要图像的挂钩

181	<code>linux_proc_maps</code>	- 收集进程内存映射
182	<code>linux_proc_maps_rb</code>	- 通过映射红黑树为 <code>linux</code> 收集进程映射
183	<code>linux_procdump</code>	- 将进程的可执行映像转储到磁盘
184	<code>linux_process_hollow</code>	- 检查进程空心的迹象
185	<code>linux_psaux</code>	- 收集进程以及完整的命令行和开始时间
186	<code>linux_psenv</code>	- 收集进程及其静态环境变量
187	<code>linux_pslist</code>	- 通过遍历 <code>task_struct->task</code> 列表来收集活动任务
188	<code>linux_pslist_cache</code>	- 从 <code>kmem_cache</code> 收集任务
189	<code>linux_psscanner</code>	- 扫描进程的物理内存
190	<code>linux_pstree</code>	- 显示进程之间的父/子关系
191	<code>linux_psxview</code>	- 使用各种进程列表查找隐藏进程
192	<code>linux_recover_filesystem</code>	- 从内存中恢复整个缓存文件系统
193	<code>linux_route_cache</code>	- 从内存中恢复路由缓存
194	<code>linux_sk_buff_cache</code>	- 从 <code>sk_buff kmem_cache</code> 中恢复数据包
195	<code>linux_slabinfo</code>	- 在运行的机器上模拟 <code>/proc/slabinfo</code>
196	<code>linux_strings</code>	- 将物理偏移量与虚拟地址匹配（可能需要一段时间，非常冗长）
197	<code>linux_threads</code>	- 打印进程的线程
198	<code>linux_tmpfs</code>	- 从内存中恢复 <code>tmpfs</code> 文件系统
199	<code>linux_truecrypt_passphrase</code>	- 恢复缓存的 <code>Truecrypt</code> 密码
200	<code>linux_vma_cache</code>	- 从 <code>vm_area_struct</code> 缓存中收集 <code>VMA</code>
201	<code>linux_volshell</code>	- 内存映像中的 <code>shell</code>
202	<code>linux_yarascan</code>	- <code>Linux</code> 内存映像中的 <code>shell</code>
203	<code>lsadump</code>	- 从注册表中转储（解密的） <code>LSA</code> 机密
204	<code>mac_adium</code>	- 列出 <code>Adium</code> 消息
205	<code>mac_apihooks</code>	- 检查进程中的 <code>API</code> 挂钩
206	<code>mac_apihooks_kernel</code>	- 检查系统调用和内核函数是否被挂钩
207	<code>mac_arp</code>	- 打印 <code>arp</code> 表
208	<code>mac_bash</code>	- 从 <code>bash</code> 进程内存中恢复 <code>bash</code> 历史记录
209	<code>mac_bash_env</code>	- 恢复 <code>bash</code> 的环境变量
210	<code>mac_bash_hash</code>	- 从 <code>bash</code> 进程内存中恢复 <code>bash</code> 哈希表
211	<code>mac_calendar</code>	- 从 <code>Calendar.app</code> 获取日历事件
212	<code>mac_check_fop</code>	- 验证文件操作指针
213	<code>mac_check_mig_table</code>	- 列出内核 <code>MIG</code> 表中的整体
214	<code>mac_check_syscall_shadow</code>	- 查找影子系统调用表
215	<code>mac_check_syscalls</code>	- 检查系统调用表条目是否被挂钩
216	<code>mac_check_sysctl</code>	- 检查未知的 <code>sysctl</code> 处理程序
217	<code>mac_check_trap_table</code>	- 检查 <code>mach</code> 陷阱表条目是否被钩住
218	<code>mac_compressed_swap</code>	- 打印 <code>Mac OS X VM</code> 压缩器统计数据并转储所有压缩页面
219	<code>mac_contacts</code>	- 从 <code>Contacts.app</code> 获取联系人姓名
220	<code>mac_dead_procs</code>	- 打印终止/取消分配的进程
221	<code>mac_dead_sockets</code>	- 打印终止/取消分配的网络套接字
222	<code>mac_dead_vnodes</code>	- 列出释放的 <code>vnode</code> 结构
223	<code>mac_devfs</code>	- 列出文件缓存中的文件
224	<code>mac_dmesg</code>	- 打印内核调试缓冲区
225	<code>mac_dump_file</code>	- 转储指定文件
226	<code>mac_dump_maps</code>	- 转储进程的内存范围，可选地包括压缩交换中的页面
227	<code>mac_dyld_maps</code>	- 从 <code>dyld</code> 数据结构中获取进程的内存映射
228	<code>mac_find_aslr_shift</code>	- 查找 <code>10.8+</code> 图像的 <code>ASLR</code> 移位值
229	<code>mac_get_profile</code>	- 自动检测 <code>Mac</code> 配置文件
230	<code>mac_ifconfig</code>	- 列出所有设备的网络接口信息
231	<code>mac_interest_handlers</code>	- 列出 <code>IOKit</code> 兴趣处理程序
232	<code>mac_ip_filters</code>	- 报告任何挂钩的 <code>IP</code> 过滤器
233	<code>mac_kernel_classes</code>	- 列出内核中加载的 <code>C++</code> 类
234	<code>mac_kevents</code>	- 显示进程的父/子关系

235	mac_keychaindump	- 恢复可能的钥匙串密钥。 使用chainbreaker打开相关的keychain文件
236	mac_ldrmodules	- 将 proc 映射的输出与 libdl 中的库列表进行比较
237	mac_librarydump	- 转储进程的可执行文件
238	mac_list_files	- 列出文件缓存中的文件
239	mac_list_kauth_listeners	- 列出 kauth Scope 监听器
240	mac_list_kauth_scopes	- 列出 kauth 范围及其状态
241	mac_list_raw	- 列出具有混杂套接字的应用程序
242	mac_list_sessions	- 枚举会话
243	mac_list_zones	- 打印活动区域
244	mac_lsmmod	- 列出加载的内核模块
245	mac_lsmmod_iokit	- 列出通过 IOkit 加载的内核模块
246	mac_lsmmod_kext_map	- 列出加载的内核模块
247	mac_lsof	- 列出每个进程打开的文件
248	mac_machine_info	- 打印有关样本的机器信息
249	mac_malfind	- 寻找可疑的进程映射
250	mac_memdump	- 将可寻址内存页转储到文件中
251	mac_moddump	- 将指定的内核扩展写入磁盘
252	mac_mount	- 打印挂载的设备信息
253	mac_netstat	- 列出每个进程的活动网络连接
254	mac_network_conns	- 列出来自内核网络结构的网络连接
255	mac_notesapp	- 查找 Notes 消息的内容
256	mac_notifiers	- 检测将钩子添加到 I/O 工具包中的 rootkit (例如 LogKext)
257	mac_orphan_threads	- 列出不映射回已知模块/进程的线程
258	mac_pgrp_hash_table	- 遍历进程组哈希表
259	mac_pid_hash_table	- 遍历 pid 哈希表
260	mac_print_boot_cmdline	- 打印内核启动参数
261	mac_proc_maps	- 获取进程的内存映射
262	mac_procdump	- 转储进程的可执行文件
263	mac_psaux	- 在用户区打印带有参数的进程 (**argv)
264	mac_psenv	- 在用户空间打印带有环境的进程 (**envp)
265	mac_pslist	- 列出正在运行的进程
266	mac_pstree	- 显示进程的父/子关系
267	mac_psxview	- 使用各种进程列表查找隐藏进程
268	mac_recover_filesystem	- 恢复缓存的文件系统
269	mac_route	- 打印路由表
270	mac_socket_filters	- 报告套接字过滤器
271	mac_strings	- 将物理偏移量与虚拟地址匹配 (可能需要一段时间, 非常冗长)
272	mac_tasks	- 列出活动任务
273	mac_threads	- 列出进程线程
274	mac_threads_simple	- 列出线程及其开始时间和优先级
275	mac_timers	- 报告内核驱动程序设置的定时器
276	mac_trustedbsd	- 列出恶意的trustedbsd 策略
277	mac_version	- 打印 Mac 版本
278	mac_vfsevents	- 列出过滤文件系统事件的进程
279	mac_volshell	- 内存映像中的外壳
280	mac_yarascan	- 扫描内存中的 yara 签名
281	machoinfo	- 转储 Mach-O 文件格式信息
282	malfind	- 查找隐藏和注入的代码
283	mbrparser	- 扫描并解析潜在的主引导记录 (MBR)
284	memdump	- 转储进程的可寻址内存
285	memmap	- 打印内存映射
286	messagehooks	- 列出桌面和线程窗口消息挂钩

287	mftparser	- 扫描并解析潜在的 MFT 条目
288	moddump	- 将内核驱动程序转储到可执行文件示例
289	modscan	- 内核模块的池扫描器
290	modules	- 打印加载模块的列表
291	multiscan	- 一次扫描各种对象
292	mutantscan	- 互斥对象的池扫描器
293	netscan	- 扫描 Vista（或更高版本）图像的连接和套接字
294	notepad	- 列出当前显示的记事本文本
295	objtypescan	- 扫描 windows 对象类型对象
296	patcher	- 基于页面扫描修补内存
297	poolpeek	- 可配置的池扫描器插件
298	pooltracker	- 显示池标签使用的摘要
299	printkey	- 打印注册表项及其子项和值
300	privs	- 显示进程权限
301	procdump	- 将进程转储到可执行文件示例
302	pslist	- 按照 EPROCESS 列表打印所有正在运行的进程
303	psscan	- 进程对象的池扫描器
304	pstree	- 将进程列表打印为树
305	psxview	- 使用各种进程列表查找隐藏进程
306	qemuinfo	- 转储 Qemu 信息
307	raw2dmp	- 将物理内存样本转换为 windbg 故障转储
308	screenshot	- 保存基于 GDI 窗口的伪截图
309	servicediff	- 列出 windows 服务 (ala Plugx)
310	sessions	- 列出 _MM_SESSION_SPACE 的详细信息 (用户登录会话)
311	shellbags	- 打印 ShellBags 信息
312	shimcache	- 解析应用程序兼容性 Shim Cache 注册表项
313	shutdowntime	- 从注册表打印机器的 ShutdownTime
314	sockets	- 打印打开的套接字列表
315	sockscan	- tcp 套接字对象的池扫描器
316	ssdt	- 显示 SSDT 条目
317	strings	- 将物理偏移量与虚拟地址匹配 (可能需要一段时间, 非常冗长)
318	svcs	- 扫描 windows 服务
319	symlinkscan	- 符号链接对象的池扫描器
320	thrdscan	- 线程对象的池扫描器
321	threads	- 调查 _ETHREAD 和 _KTHREADS
322	timeliner	- 从内存中的各种工件创建时间线
323	timers	- 打印内核定时器和相关的模块 DPC
324	truecryptmaster	- 恢复 TrueCrypt 7.1a 主密钥
325	truecryptpassphrase	- TrueCrypt 缓存密码短语查找器
326	truecryptsummary	- TrueCrypt 总结
327	unloadedmodules	- 打印已卸载模块的列表
328	userassist	- 打印 userassist 注册表项和信息
329	userhandles	- 转储 USER 句柄表
330	vaddump	- 将 vad 部分转储到文件中
331	vadinfo	- 转储 VAD 信息
332	vadtree	- 遍历 VAD 树并以树格式显示
333	vadwalk	- 走 VAD 树
334	vboxinfo	- 转储 virtualbox 信息
335	verinfo	- 从 PE 图像中打印出版本信息
336	vmwareinfo	- 转储 VMware VMSS/VMSN 信息
337	volshell	- 内存映像中的 shell
338	win10cookie	- 查找 windows 10 的 ObHeaderCookie 值
339	windows	- 打印桌面窗口 (详细信息)
340	wintree	- 打印 Z 顺序桌面 windows 树

341	wndscan	- 用于窗口站的池扫描仪
342	yarascan	- 使用 Yara 签名扫描进程或内核内存
343		
344		
345	Scanner Checks	
346	-----	
347	CheckPoolSize	- 检查池块大小
348	CheckPoolType	- 检查池类型
349	KPCRScannerCheck	- 检查自引用指针以查找KPCR
350	MultiPrefixFinderCheck	- 每页检查多个字符串，在偏移处完成
351	MultiStringFinderCheck	- 每页检查多个字符串
352	PoolTagCheck	- 此扫描程序检查池标记的出现