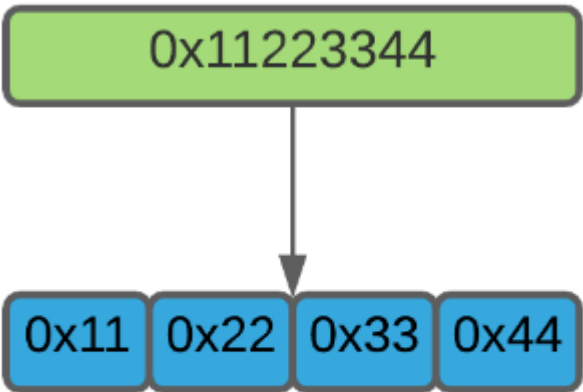


x64

Shellcode

CPU CPU:1 C

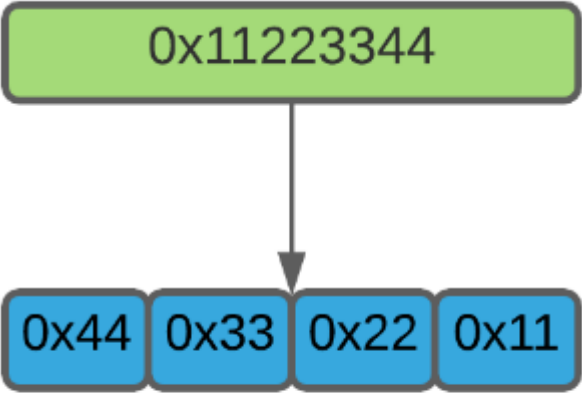
()



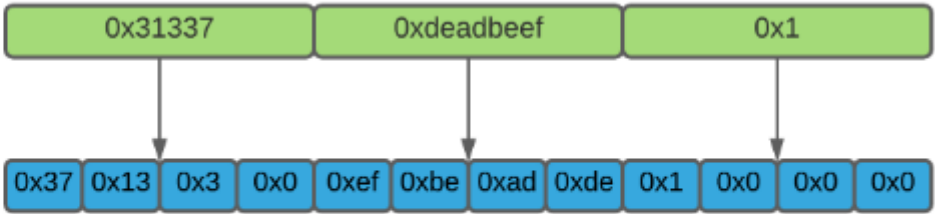
Memory Address: 0x0 0x1 0x2 0x3

0x11223344 4 0x11 0x22 0x33 0x44

0x11223344 4 0x11 0x22 0x33 0x44



Memory Address: `0x0` `0x1` `0x2` `0x3`



Memory Address: `0x0` `0x1` `0x2` `0x3` `0x4` `0x5` `0x6` `0x7` `0x8` `0x9` `0xa` `0xb`

$2^{64}-1$ -2^{63} $2^{63}-1$ 0

$1 \ll 42$ 2

42	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0010	1010
	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1101	0101
1:	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1101	0110

CPU

(RAM) CPU 64 8 64 x64

RIP		
RAX	I/O	Windows syscall SSN
RBX		
RCX		
RDX	I/O	
RSI		
RDI		
RBP		
RSP		
R8-R15		
RFLAGS	FLAG	

x64

8

RAX

4

EAX

EAX

2

AX

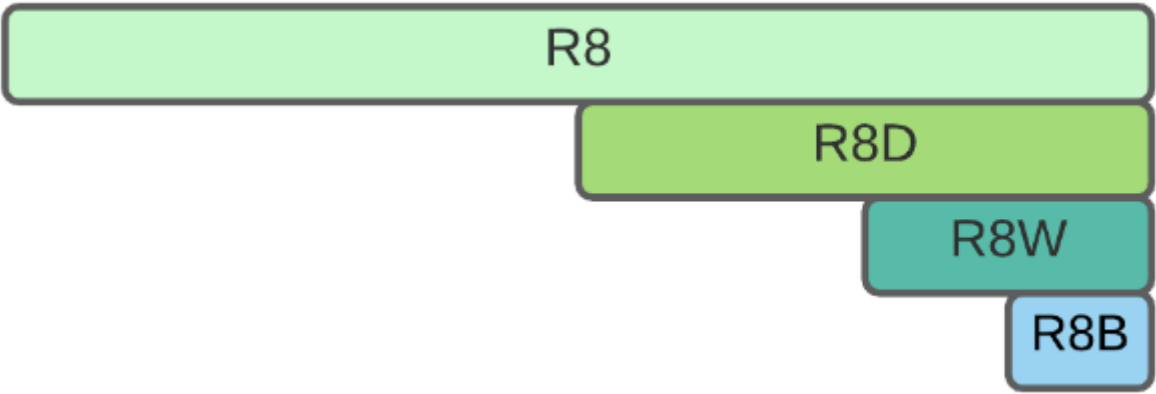
AH

AL

A



R8



32 EAX 32 0 8 16

64 ???? ?	? 32 ?	? 16 ?	? 8 ?
rax	eax	ax	al
rbx	ebx	bx	bl
rcx	ecx	cx	cl
rdx	edx	dx	dl
rsi	esi	si	sil
rdi	edi	di	dil
rbp	ebp	bp	bpl
rsp	esp	sp	spl
r8	r8d	r8w	r8b
r9	r9d	r9w	r9b

64 ???? ?	32 ?	16 ?	8 ?
r10	r10d	r10w	r10b
r11	r11d	r11w	r11b
r12	r12d	r12w	r12b
r13	r13d	r13w	r13b
r14	r14d	r14w	r14b
r15	r15d	r15w	r15b

RFLAGS

0	CF	
2	PF	
6	ZF	
7	SF	
11	OF	

(LIFO)

Windows x64 (Linux x64)	Windows x86	RCX	RDX	R8	R9	
						4

x64 Windows	RCX	RDX	R8	R9
-------------	-----	-----	----	----

RAX R10 R11 R12 R13 R14 R15 R16 R17 R18 R19 R20 R21 R22 R23 R24 R25 R26 R27 R28 R29 R30 R31 R32 R33 R34 R35 R36 R37 R38 R39 R40 R41 R42 R43 R44 R45 R46 R47 R48 R49 R50 R51 R52 R53 R54 R55 R56 R57 R58 R59 R60 R61 R62 R63 R64 R65 R66 R67 R68 R69 R70 R71 R72 R73 R74 R75 R76 R77 R78 R79 R80 R81 R82 R83 R84 R85 R86 R87 R88 R89 R90 R91 R92 R93 R94 R95 R96 R97 R98 R99 R100 R101 R102 R103 R104 R105 R106 R107 R108 R109 R110 R111 R112 R113 R114 R115 R116 R117 R118 R119 R120 R121 R122 R123 R124 R125 R126 R127 R128 R129 R130 R131 R132 R133 R134 R135 R136 R137 R138 R139 R140 R141 R142 R143 R144 R145 R146 R147 R148 R149 R150 R151 R152 R153 R154 R155 R156 R157 R158 R159 R160 R161 R162 R163 R164 R165 R166 R167 R168 R169 R170 R171 R172 R173 R174 R175 R176 R177 R178 R179 R180 R181 R182 R183 R184 R185 R186 R187 R188 R189 R190 R191 R192 R193 R194 R195 R196 R197 R198 R199 R200 R201 R202 R203 R204 R205 R206 R207 R208 R209 R210 R211 R212 R213 R214 R215 R216 R217 R218 R219 R220 R221 R222 R223 R224 R225 R226 R227 R228 R229 R230 R231 R232 R233 R234 R235 R236 R237 R238 R239 R240 R241 R242 R243 R244 R245 R246 R247 R248 R249 R250 R251 R252 R253 R254 R255 R256 R257 R258 R259 R260 R261 R262 R263 R264 R265 R266 R267 R268 R269 R270 R271 R272 R273 R274 R275 R276 R277 R278 R279 R280 R281 R282 R283 R284 R285 R286 R287 R288 R289 R290 R291 R292 R293 R294 R295 R296 R297 R298 R299 R300 R301 R302 R303 R304 R305 R306 R307 R308 R309 R310 R311 R312 R313 R314 R315 R316 R317 R318 R319 R320 R321 R322 R323 R324 R325 R326 R327 R328 R329 R330 R331 R332 R333 R334 R335 R336 R337 R338 R339 R340 R341 R342 R343 R344 R345 R346 R347 R348 R349 R350 R351 R352 R353 R354 R355 R356 R357 R358 R359 R360 R361 R362 R363 R364 R365 R366 R367 R368 R369 R370 R371 R372 R373 R374 R375 R376 R377 R378 R379 R380 R381 R382 R383 R384 R385 R386 R387 R388 R389 R390 R391 R392 R393 R394 R395 R396 R397 R398 R399 R400 R401 R402 R403 R404 R405 R406 R407 R408 R409 R410 R411 R412 R413 R414 R415 R416 R417 R418 R419 R420 R421 R422 R423 R424 R425 R426 R427 R428 R429 R430 R431 R432 R433 R434 R435 R436 R437 R438 R439 R440 R441 R442 R443 R444 R445 R446 R447 R448 R449 R450 R451 R452 R453 R454 R455 R456 R457 R458 R459 R460 R461 R462 R463 R464 R465 R466 R467 R468 R469 R470 R471 R472 R473 R474 R475 R476 R477 R478 R479 R480 R481 R482 R483 R484 R485 R486 R487 R488 R489 R490 R491 R492 R493 R494 R495 R496 R497 R498 R499 R500 R501 R502 R503 R504 R505 R506 R507 R508 R509 R510 R511 R512 R513 R514 R515 R516 R517 R518 R519 R520 R521 R522 R523 R524 R525 R526 R527 R528 R529 R530 R531 R532 R533 R534 R535 R536 R537 R538 R539 R540 R541 R542 R543 R544 R545 R546 R547 R548 R549 R550 R551 R552 R553 R554 R555 R556 R557 R558 R559 R560 R561 R562 R563 R564 R565 R566 R567 R568 R569 R570 R571 R572 R573 R574 R575 R576 R577 R578 R579 R580 R581 R582 R583 R584 R585 R586 R587 R588 R589 R590 R591 R592 R593 R594 R595 R596 R597 R598 R599 R600 R601 R602 R603 R604 R605 R606 R607 R608 R609 R610 R611 R612 R613 R614 R615 R616 R617 R618 R619 R620 R621 R622 R623 R624 R625 R626 R627 R628 R629 R630 R631 R632 R633 R634 R635 R636 R637 R638 R639 R640 R641 R642 R643 R644 R645 R646 R647 R648 R649 R650 R651 R652 R653 R654 R655 R656 R657 R658 R659 R660 R661 R662 R663 R664 R665 R666 R667 R668 R669 R670 R671 R672 R673 R674 R675 R676 R677 R678 R679 R680 R681 R682 R683 R684 R685 R686 R687 R688 R689 R690 R691 R692 R693 R694 R695 R696 R697 R698 R699 R700 R701 R702 R703 R704 R705 R706 R707 R708 R709 R710 R711 R712 R713 R714 R715 R716 R717 R718 R719 R720 R721 R722 R723 R724 R725 R726 R727 R728 R729 R730 R731 R732 R733 R734 R735 R736 R737 R738 R739 R740 R741 R742 R743 R744 R745 R746 R747 R748 R749 R750 R751 R752 R753 R754 R755 R756 R757 R758 R759 R760 R761 R762 R763 R764 R765 R766 R767 R768 R769 R770 R771 R772 R773 R774 R775 R776 R777 R778 R779 R780 R781 R782 R783 R784 R785 R786 R787 R788 R789 R790 R791 R792 R793 R794 R795 R796 R797 R798 R799 R800 R801 R802 R803 R804 R805 R806 R807 R808 R809 R810 R811 R812 R813 R814 R815 R816 R817 R818 R819 R820 R821 R822 R823 R824 R825 R826 R827 R828 R829 R830 R831 R832 R833 R834 R835 R836 R837 R838 R839 R840 R841 R842 R843 R844 R845 R846 R847 R848 R849 R850 R851 R852 R853 R854 R855 R856 R857 R858 R859 R860 R861 R862 R863 R864 R865 R866 R867 R868 R869 R870 R871 R872 R873 R874 R875 R876 R877 R878 R879 R880 R881 R882 R883 R884 R885 R886 R887 R888 R889 R890 R891 R892 R893 R894 R895 R896 R897 R898 R899 R900 R901 R902 R903 R904 R905 R906 R907 R908 R909 R910 R911 R912 R913 R914 R915 R916 R917 R918 R919 R920 R921 R922 R923 R924 R925 R926 R927 R928 R929 R930 R931 R932 R933 R934 R935 R936 R937 R938 R939 R940 R941 R942 R943 R944 R945 R946 R947 R948 R949 R950 R951 R952 R953 R954 R955 R956 R957 R958 R959 R960 R961 R962 R963 R964 R965 R966 R967 R968 R969 R970 R971 R972 R973 R974 R975 R976 R977 R978 R979 R980 R981 R982 R983 R984 R985 R986 R987 R988 R989 R990 R991 R992 R993 R994 R995 R996 R997 R998 R999 R1000 R1001 R1002 R1003 R1004 R1005 R1006 R1007 R1008 R1009 R1010 R1011 R1012 R1013 R1014 R1015 R1016 R1017 R1018 R1019 R1020 R1021 R1022 R1023 R1024 R1025 R1026 R1027 R1028 R1029 R1030 R1031 R1032 R1033 R1034 R1035 R1036 R1037 R1038 R1039 R1040 R1041 R1042 R1043 R

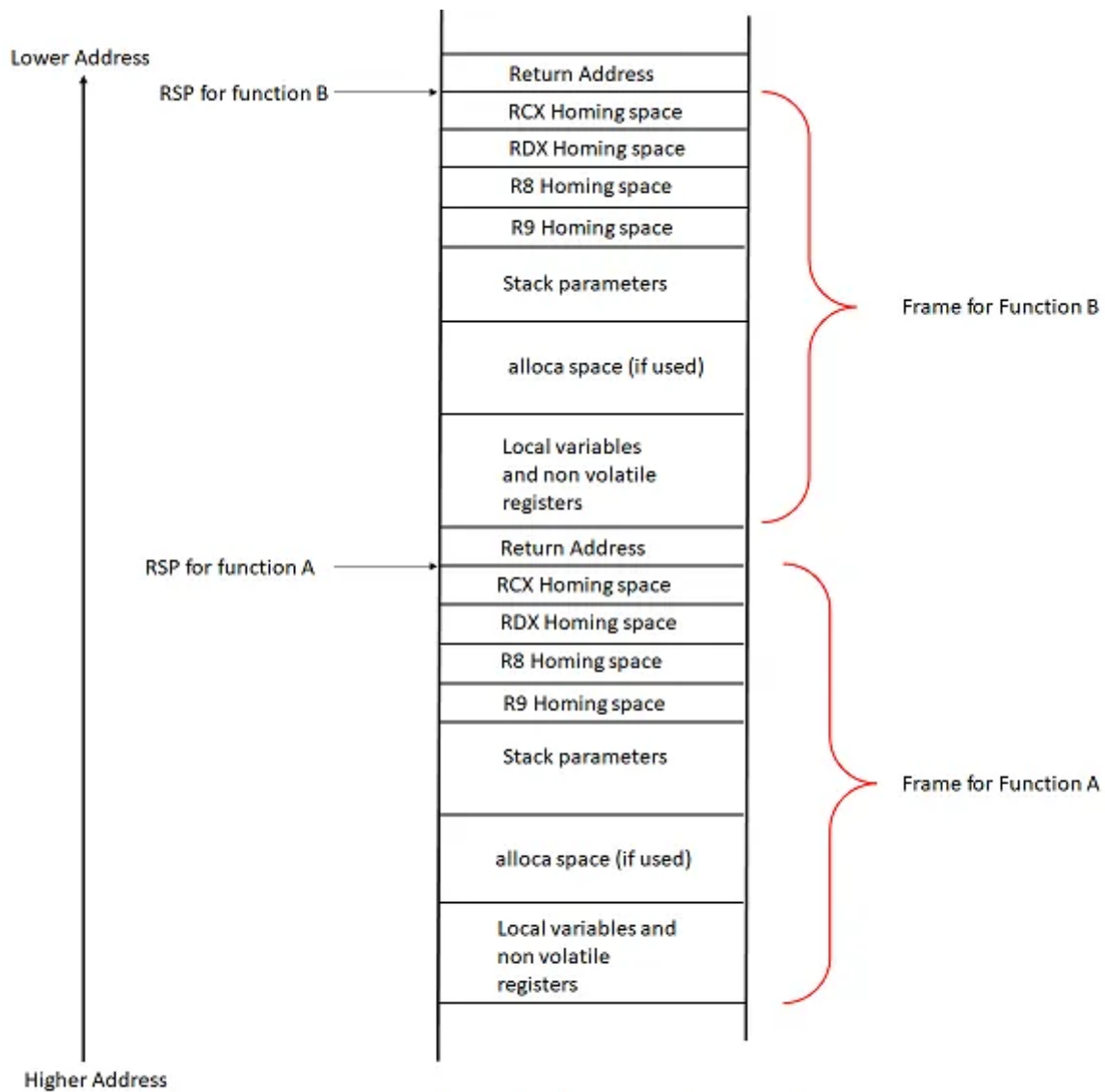
```
RBX RBP RDI RSI RSP R12 R13 R14 R15
```

x86-64 fastcall PUSH/POP RSP

) (RIP

(RSP)	RSP	RSP	
(RBP)		x64	RSP

A	B
1	1
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
1	10
1	11
1	12
1	13
1	14
1	15
1	16
1	17
1	18
1	19
1	20
1	21
1	22
1	23
1	24
1	25
1	26
1	27
1	28
1	29
1	30
1	31
1	32
1	33
1	34
1	35
1	36
1	37
1	38
1	39
1	40
1	41
1	42
1	43
1	44
1	45
1	46
1	47
1	48
1	49
1	50
1	51
1	52
1	53
1	54
1	55
1	56
1	57
1	58
1	59
1	60
1	61
1	62
1	63
1	64
1	65
1	66
1	67
1	68
1	69
1	70
1	71
1	72
1	73
1	74
1	75
1	76
1	77
1	78
1	79
1	80
1	81
1	82
1	83
1	84
1	85
1	86
1	87
1	88
1	89
1	90
1	91
1	92
1	93
1	94
1	95
1	96
1	97
1	98
1	99
1	100



Stack frame when function A calls function B

BYTE	1
WORD	2
DWORD	4
QWORD	8

Shellcode

MOV

MOV

```
MOV RAX, 1 // 1 RAX
MOV [RAX], 3 // 3 RAX
MOV RAX, RCX // RCX RAX
MOV [RDI], RAX // RAX RDI
MOV [RAX], RAX // RAX RAX
MOV RBX, [RDI + 0x10] // RDI 10 RBX
```

LEA

LEA MOV

```
LEA RBX, [RCX + 0x10] // RCX 10 RBX
MOV RBX, [RCX + 0x10] // RCX 10 RBX
LEA RAX, [RCX + 2*RAX + 0x10] // RCX + 2*EAX + 10 RAX
```

PUSH/POP

PUSH POP x64 x64 PUSH POP

```
PUSH RAX // RAX
PUSH 1 // 1
POP RAX // RAX
```

INC/DEC/ADD/SUB/MUL/DIV

[illegible]

```
INC RAX      // RAX = RAX + 1
INC BYTE [RAX] // RAX = RAX + 1
ADD RAX, RAX  // RAX = RAX + RAX
ADD RCX, 4     // RCX = RCX + 4
ADD DWORD [RSP], RAX // memory[RSP] = memory[RSP] + RAX
SUB RAX, RDX   // RAX = RAX - RDX
SUB RBX, 0x10  // RBX = RBX - 0x10
MUL RCX        // RDX: RAX = RAX * RCX
MUL DWORD [RDX] // RDX: RAX = RAX * memory[RDX]
DIV RCX        // RDX: RAX/RCX=RAX··RDX
```

NEG

NEG	0x00
-----	------

```
NEG RAX // RAX = -RAX
```

AND/OR/XOR/NOT

$$0 \text{ AND } 0 = 0$$
$$0 \text{ AND } 1 = 0$$
$$1 \text{ AND } 0 = 0$$
$$1 \text{ AND } 1 = 1$$
$$0 \text{ OR } 0 = 0$$
$$0 \text{ OR } 1 = 1$$
$$1 \text{ OR } 0 = 1$$
$$1 \text{ OR } 1 = 1$$

$$1 \text{ XOR } 1 = 0$$

NOT 1 = 0

NOT **ADD RAX, RCX**

[illegible]

AND RCX, 0x11 ; RCX = RCX and 0x11

CALL RET

jxx

JNE/JNZ / 0

JE/JZ	/ 0
JNE/JNZ	0
JG/JNLE	/
JGE/JNL	/
JL/JNGE	/
JLE/JNG	/

SHL SAL SHR SAR () SAR

2 1 2^n

[illegible]

1111 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

ROL/ROR " "

```

┌───────────┐
1 0 0 0 1 0 0 0 |
  / / / / / / / |
0 0 0 1 0 0 0 1 └─┘

```

STOS	BYTE	WORD	DWORD	QWORD	RDI	1	RDI	1	QWC
STOS REP	CRCX	STOSRCX	1	0	REP STOS	memset			
SCAS	(AL AX EAX RAX)					DF	RDI	SCAS	RI

Revision #27
Created 1 May 2023 13:41:06 by
Updated 28 January 2024 05:09:29 by