

US Army Cyber School

2018

Reverse Engineering - Activity



x64 Stack Practice

Introduction

In this activity, you will be stepping through x64 ASM source code. You will need to show everything that is happening on the stack. As well as, how the registers are being affected. The following source code will be broken down over the next few pages:

main:

```
mov rax, 5
push rax
mov rax, 1
push rax
pop r12
pop r13
```

loop:

```
add r12, 1
cmp r12, r13
jl loop
```

```
mov rax, 0
ret
```

****The first portion of this code on page 3 has been completed for you as an example.****

Activity

main:

```
mov rax, 5
```

Registers

rax = 5

Stack

RSP

Notes

5 is moved into rax.
The rsp remains at the
highest position in memory
on the stack.

rsp

Activity

[illegible]

Activity

[illegible]

Activity

Registers

Stack

RSP

Notes

Activity

RSP[illegible][illegible]

Patient Information	
First Name	
Last Name	
Address	
City	
State	
Zip	
Phone	
Insurance	
Physician Information	
Physician Name	
Physician Address	
Physician City	
Physician State	
Physician Zip	
Physician Phone	
Physician Insurance	
Referral Information	
Referral Number	
Referral Date	
Referral Type	
Referral Source	
Referral Reason	
Referral Status	
Referral Notes	
Referral History	
Referral Date	
Referral Type	
Referral Source	
Referral Reason	
Referral Status	
Referral Notes	
Referral Summary	
Referral Date	
Referral Type	
Referral Source	
Referral Reason	
Referral Status	
Referral Notes	

Activity

Registers

Stack

RSP

Notes

Activity

```
loop:  
    add r12, 1  
    cmp r12, r13  
    jl loop
```

Provide an explanation of what is happening in this code segment to include how the registers are affected and what actions are taken.

Activity

[illegible]

Activity

Registers

Stack

RSP

Notes