Verkefni 1 skýrsla

hir12

October 2022

Phase 1

All your base are belong to us.
Þetta fann ég út hreinlega bara með því að prenta út hex strenginn sem að kemur í forritinu. Það er voða lítið meira sem er hægt að segja um phase 1 nema það að ég prófaði að keyra skipunina: p (char*) 0x555555557150 sem að prentar út strenginn.

Phase 2

1 3 9 27 81 243
Það fyrsta sem að ég sé hér er að inntakið eru 6 tölur. Ég sé það á línunni:

```
call    0x555555555d0b <read_six_numbers>
```

Þá prófa ég hreinlega að setja inn einhverjar 6 tölur og sjá hvar ég stoppa
Þá sé ég að í línunni

```
cmpl    $0x1, (%rsp)
```

þarf að koma 1 í fyrstu tölunni svo að hún fari ekki í stökkið og sprengi.
Næst sé ég að 3 línur hafa mikil áhrif á næstu skref, þær eru:

```
lea     (%rax, %rax, 2), %eax
cmp     %eax, 0x4(%rbx)
je      # hopp aftur upp
```

Það sem þetta er að gera er að lea skipunin setur rax í 3 veldi í eax og svo er eax borið saman við intakið, ef satt þá er hoppað aftur upp og þetta endurtekið fyrir restina af tölunum.
Þegar forritið er komið í gegnum þessar 5 tölur þá fer það í hoppið:

```
cmp     %rbp, %rbx
je      # hoppar yfir lykkjuna
```

Eftir þetta hopp er bara tekið til í forritinu og skilað gildum.

Phase 3

0 X 444 Hér er byrja ég á því að prenta út strengina sem að eru í dæminu og fæ þar út að inntakið er tala, stafur, tala. Svo að ég prófa að setja inn eitthverjar 2 tölur og 1 staf og byrja að prófa.
Ég sé á þessum línum að verið er að athuga hvort talan sé minni en 7 svo að þá vitum við að það er eitt af skilyrðum fyrir fyrstu töluna.

```
lea     0x14(%rsp),%r8
cmp     $0x2, %eax
cmpl    $0x7, 0x10(%rsp)
```

Hér sjáum við að fyrsta talan þarf að vera minni en 7. Ég slysaðist á það að finna það út að talan væri 0.

```
cmpl    $0x1bc, 0x14(%rsp)
```

Hér sé ég að hexstrengurinn 1bc er nauðsynlegur til að sprengja ekki og næsta inntak er char svo að ég finn ascii gildið af 1bc og fæ stafinn 'X' það er næsta inntak hjá mér. Ég færi þetta sama gildi svo einnig í decimal og fæ út síðustu töluna mína 444.