Bomblab

Arnar Sigurðsson

Sprengjan mín var **bomb105** og lausnirnar sem ég fann í fyrstu 5 hlutum af bomblab verkefninu voru eftirfarandi:

1. When a problem comes along, you must zip it!
2. 1 4 9 16 25 36
3. 6 W 735
4. 10 5
5. 64 48

Ég fann þessar lausnir með því að nota gdb-aflúsaran og fara skref fyrir skref í gegnum kóðan auk þess að skoða objdump af forritinu.

Fyrsti hlutinn var mjög einfaldur en strax í annarri skipun er kallað á „mov $0x402380, %esi“ og með því að athuga hvað gildið í hólfi 0x402380 með því að gera „x /s 0x402380“ kom í ljós að þar var strengurinn „When a problem comes along, you must zip it!”. Svo strax á eftir er kallað á strings_not_equal, gerð prufun á útkomunni og ef það er ekki jafnt, jne=jump if not equal, þá er hoppað í skipun 400eb9 sem er sprengja. Þá var bara að keyra forritið aftur, slá inn strenginn og athuga hvort hann virkar, sem hann gerði.

Í hluta tvö kom framm í fimmtu skipun kall á fallið read_six_numbers sem gaf sterklega til kynna að 6 tölur hefðu eitthvað með þetta að gera. Reyndar eyddi ég svolitlum tíma í að reyna að rekja mig í gegnum fallið sjálft en áttaði mig svo á því að það er í raun óþarfi og prufaði að keyra bara phase_2 með 6 heiltölum. Strax eftir þetta kall bar það saman (%rsp) og $1, og með því að athuga hvað var í (%rsp) og sjá að það var fyrsta talan í inntakinu sá ég að sú tala átti greinilega að vera 1. Næsta compare skipun athugar hvort %eax sem var þá 4 væri jafnt og (%rbp) sem var önnur talan sem ég setti inni og þá á hún greinilega að vera 4. Svo hoppar þetta upp og gerir sömu skipanir og það geri áður svo þarna virtist maður vera kominn inn í lykkju. Í lykkjunni eru gerðar einhverjar reikniaðgerðir og svo var önnur compare skipun og í þetta skiptið var verið að bera þriðja inntakið við 9. Þegar ég var kominn hingað áttaði ég mig á að þetta virtist vera runan $1^2 - 2^2 - 3^2 - 4^2 - 5^2 - 6^2$ og eftir að prufa að setja það sem inntak komst ég í gegn.

Í þriðja hluta er fyrsta compare skipunin að bera saman %eax við 2 þar sem %eax var fjöldi inntaka af skipanalínu, ef þau voru 2 eða færri var hoppað yfir í sprengju svo greinilega áttu inntökin að vera fleiri en 2. Svo var borið saman fyrsta inntak og 7 þar sem inntakið mátti ekki vera hærra en 7. Næst var hoppað á mismunandi staði í kóðanum eftir því hvað inntakið var, eins og verið væri að fara eftir hopptöflu í switch-setningu. Ég hélt mig við töluna sem ég valdi ,6 ,til að halda henni eins og finna út frá því hvað restin átti að vera, eða ef það væri hægt yfir höfuð. Næsta compare var 0x2df eða 735 borið saman við þriðja inntakið svo að þar með hlaut það að eiga vera 735. Þegar ég var að skoða innihald gistanna rakst ég á innihaldið “%d %c %d” sem þýddi væntanlega að inntak 2 væri af taginu char. Næsta samanburðarskipun var á %al eða fyrsta bætinu í %rax og inntaki 2. í %rax var talan 0x57 eða 87 svo ég leitaði að hvaða tákn væri með númerið 87 í ASCII-kóðanum sem var W og eftir að prufa það komst ég í gegn.

Í fjórða hluta er first athugað hvort inntökin séu 2, svo er athugað hvort fyrra viðfangið er minna en eða jafnt og 14. Svo er kallað á fallið func4 sem gerir alls konar reikniaðgerðir og æfingar. Meðal annars er einu sinni kallað á shift right um 31 sæti, eins og verið sé að athuga formerkisbitann, svo er 14 deilt með 2 með shift-arithmetic right. Þetta fall fer í einhverja hringi en ég náði ekki alveg að fylgja því hvað fallið er í raun að reyna að gera, þetta virtist vera einhvers konar lykkja sem var að nota breyturnar aftur og aftur. Lykillinn virtist vera að eftir fallið átti að vera 5 í %rax og með því að prufa alls konar lausnir þar sem fyrri talan var 14 eða minni lenti ég á 10 og 5 sem hleypti mér í gegn.

Í fimmta hluta kom fram í einni skipun að inntakið átti að vera “%d %d”, svo var skannað og athugað hvort inntökin væru jöfn eða færri en 1, og ef ekki var haldið áfram. Næst var athugað hvort það sem var í %rax væri jafnt og 15 en ef svo var var sprengjan sprengd. Svo var upphafsstillt nokkur gildi fyrir það sem virtist vera lykkja. Þessi lykkja er frekar furðuleg því hún virðist vera að ná í gildi úr fylki á vistfanginu þar sem fylkið er og svo %rax+4 ofan á það. Það er svo notað sem index í fylkið þegar kallað er á það í næstu umferð. Á meðan þetta gerist er teljarinn rdx að hækka um 1 og til að komast út úr loopunni á %rax að vera jafnt og 15. Athugavert er á eftir lykkjunni verður teljarinn að vera 6, semsagt að lykkjan endar þegar %rax er 15 og þá verður teljarinn að vera kominn upp í 6 til að geta klárað hlutann. Einnig á seinna gildið úr inntakinu (48) að vera jafnt því sem kemur úr %rcx sem í þessu tilfelli var einmitt 48. Ég var að spá í hver seinni talan gæti verið og datt í hug að prufa 2^6 veldi út af sexunni sem var þarna og með algjörri heppni virkaði það. Svo prufaði ég 2^5=32 og það virkaði líka og 2^4 en 2^3 virkaði ekki en þetta virðist hafa eitthvað að gera með 2 í einhverju veldi.

Heilt yfir gekk verkefnið mjög vel og var í rauninni mjög ávanabindandi. Ég myndi segja að ég skildi þetta nokkuð vel upp að hluta 3 en í 4 og 5 þá fór þetta að verða það flókið að ég ekki alveg fylgt þræðinum, og þurfti að fara að prufa og giska mig áfram meira en áður.