

# Řešení úlohy č. 1

## Bitva v paměti

### High level možnosti řešení

Způsobů jak vyřešit tuto úlohu bylo několik. Hodnocení záleželo na tom, jak dlouho člověk zvládl přežít v porovnání s ostatními. Neexistovalo žádné správné řešení, a ani zde žádné nebude uvedeno.

Šlo tedy buď zabít ostatní, nebo se prostě snažit udržet naživu. Druhé řešení typicky nepřineslo autorovi maximální počet bodů, protože nezvládlo zabít poslední program, i když i takové případy se našly, kde program který jinak zabil všechny ostatní nakonec zabil sám sebe.

Účastníci volili obě možnosti, tu útočnou ale zvolila drtivá většina. Nenašel jsem řešení, kde by účastník skočil o 0x100 (nebo víc) instrukcí dopředu a začal vykonávat kód jiného hráče, i když jsem takové řešení očekával. Bylo by zajímavé sledovat, jak by to dopadlo - za předpokladu, že by se takový kód nějak zvládl dostat přes PVE.

### Loop condition evasion

Dle zadání byl zabít jakýkoliv program, který se zacyklil - tedy sekvenci těch stejných instrukcí (bytově) na stejných adresách spustil třikrát za sebou. Tomu se šlo relativně jednoduše vyhnout pomocí instrukce BOMB. Tato instrukce, ač tak na začátku mohla vypadat, nesloužila k zabíjení hráčů - když jste na jejich pozici zapsali instrukci BOMB, stejně tak jste tam mohli zapsat nulu, se stejným efektem, ale kratším kódem. V zadání je ale napsáno, že instrukce BOMB se při každém spuštění zmodifikovala - v paměti se změnila tak, že se timer o jedničku zkrátil. Toho šlo využít, protože kód vykonávající v cyklu instrukci BOMB měnil svůj obsah v paměti. Díky tomu nebyla splněna podmínka pro zacyklení, a program nebyl zabít.<sup>1</sup>

### Zabíjení hráčů

Jeden ze způsobů zabíjení hráčů bylo přepsat jejich vlastní kód. K tomu bylo potřeba zjistit kde hráč je, a to pomocí čtení z paměti na adresách 42 a 43. Adresa 42 obsahovala adresu nejbližšího programu - tedy ne nejbližšího oponenta, ale vaši vlastní! Na to se nacyklovalo několik hráčů. Adresa 43 byla ta, která je skutečně užitečná.

Na adresu, kterou takto získáme, je možné pomocí psaní do paměti zapsat kód (resp. raději na místo o kousek dál, než na kterou adresu ukazovala - než tam zapíšeme, hráč už bude o kousek jinde). Nejefektivnější bylo zapsat tam prostě nulu, ale našli se hráči, kteří do zapisovaných bytů dávali svůj podpis - pár bytů, které nic neovlivnili, ale jejich oběť věděla, kdo za to vlastně mohl.

Druhý ze způsobů byl teleport - nakonec se ale jednalo o instrukci, která téměř nebyla využitelná. Když byl zrovna hráč suspendnutý v teleportu (což je jinak dobrý způsob jak přežít dlouho), mohl být zabít tím, že někdo změnil kód toho samotného teleportu. Nemálo vítězných řešení na tuto možnost myslelo a danou adresu hráče vrácenou z adresy 43 přepisovali i bez offsetu. Z důvodu malé praktičnosti tak nebyl téměř vůbec použitý v praxi.

<sup>1</sup>Mohli jste si všimnout, že počítače, proti kterým jste bojovali v sobě instrukci BOMB nemají, a přesto neumrou kvůli zacyklení. Jedná se o výjimku v běhovém prostředí, abych neukázal žádný ze způsobů, jak se kontrole zacyklení vyhnout.

## Utíkání před hráči

Řada hráčů dříve či později zjistila, že je potřeba nahodile skákat, aby se vyhla nástrahám jiných hráčů. Nejčastější byl skok o nějaký počet instrukcí dopředu, třeba kolem deseti, ale ke konci hry řada hráčů skákala pozadu. Nejdříve skočili úplně dozadu v programu, a následně doskákali skoro na začátek – a to klidně i instrukci po instrukci – a buď v průběhu skákání, nebo přímo u začátku, spustili samotný kód na zabíjení hráčů. Skákání dozadu byl definitivně nejefektivnější způsob, jak přežít, i když ti nejlepší hráči dávali nástrahy i před zjištěnou adresu hráče.

Druhá možnost, byť velmi výrazně nepraktická, bylo čtení vlastního kódu a jeho následná oprava. Tento způsob jsem u jednoho hráče našel, byť nebyl moc praktický ani použitelný – zkrátka na to nebyly dostatečné možnosti.

## Statistiky

Celkově se zaregistrovalo 66 hráčů, z nich se finální bitvy mohlo účastnit 54.

### Okamžité útoky

Hráči ve finální bitvě použili jako první instrukci:

- 20-krát 05 – LOAD
- 16-krát 10 – JUMP
- 5-krát 21 – SETIMMHIGH
- 3-krát 20 – SETIMMLOW
- 2-krát 07 – MOV
- 1-krát 69 a 50 – NOP a BOMB

Okamžitě tedy předpokládám, že začalo útočit 30 hráčů, zhruba 16 začalo okamžitě skákat pryč.

### Špatné načítání adresy hráče

Adresu 42 zkusilo načíst celkem 288 odevzdaných programů od 34 hráčů, zatímco adresu 43 zkusilo načíst 408 odevzdání (z celkových 714) od 49 hráčů.

### Teleport

Teleport zkusilo použít 180 odevzdání od 24 různých hráčů. Ve finální bitvě teleport použilo 13 hráčů, což je víc, než bych sám čekal.

### Bomba

Bombu použilo celkově 252 odevzdání od 32 hráčů, ve finální bitvě byla použita u pouze 24 hráčů. Jsem překvapený že ji více lidí nepoužilo pro únik od podmínky zacyklení, nicméně je pravda, že ty nejúspěšnější programy ji vůbec nepotřebovaly – útočily sice zřídka, ale úspěšně, a mezi útoky měly velké odstupy plné skoků sem a tam po programu, tedy hra zpravidla stihla doběhnout než je zacyklení zabilo – nebo alespoň přežili dost dlouho na získání pro ně dostatečného počtu bodů-

Celkově jsem si úlohu jako organizátor moc užil a těší mě, kolik vynalézavosti jsem viděl u účastníků. Jestli bude čas i nápad, tak bych moc rád podobný typ úlohy ještě někdy zopakoval v dalších ročnících. Viděl jsem použité techniky, které mě samotného na začátku nenapadly a jsem moc rád, že to tak dopadlo.

Dobrá práce! <sup>2</sup>

---

<sup>2</sup>Pro případ že byste si ještě někdy chtěli tenhle jazyk vyzkoušet, je to open source: <https://github.com/SoptikHa2/fiks-cpu>