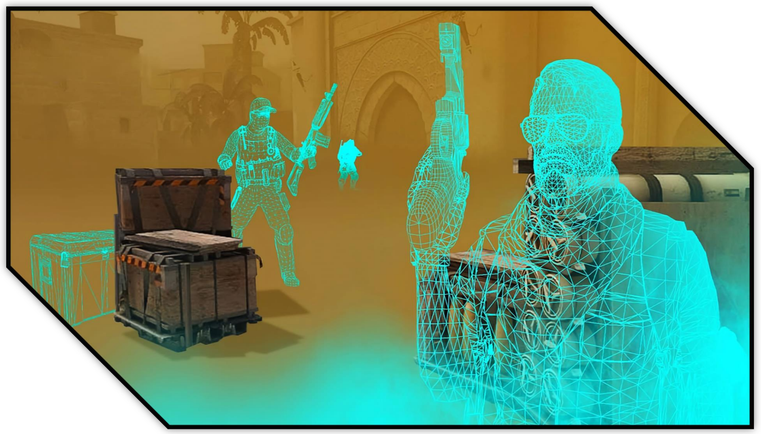
Reverse Engineering Shenanigans

Overview

The aim of this project is to both improve and leverage the reverse engineering skills on a game in order to understand its internal structure, as well as offer an advantage for the players that are looking for a fast improvement. The targeted game is CS:GO, which is a competitive team-based first-person shooter.

Features

The software fetches data and modifies the game's memory in order to offer functionality such as AimBot, ESP, TriggerBot, BunnyHop, and more. These features greatly assist and increase the player's performance. On top of that, the software uses memory scanning to get the relevant addresses in case of a game update.



How was it made?

The whole process involves a lot of trial and error. The game has been reversed using various tools to ease the job. The Interactive Disassembler was used to disassemble the game's binaries for static analysis. However, for debugging and real-time analysis, tools such as 'Cheat Engine' and 'ReClass.NET' came in handy.

```
HudProcessInput proc near               ; DATA XREF: .rdata:36E
                push    ebp
                mov     ebp, esp
                mov     ecx, pIClientModePtr
                mov     eax, [ecx]
                pop     ebp
                jmp     dword ptr [eax+48] ; Indirect Near Jump

HudProcessInput endp
```

How does it work?

The software is an injectable DLL, therefore, it runs seamlessly as if it were a part of the game.

To execute its own code, the DLL hooks/hijacks the game's code and redirects the execution flow to the DLL's code.

Is it ethical?

It is a controversial topic because, at the end of the day, the software offers an unfair advantage. Civil utilization of the software is welcome. This includes use cases such as 'Singleplayer', 'Hack vs Hack', and features that do not negatively impact other players' experience.
However, the use of it for exploitation and malicious intents is always at the user's conscience and moral standards.

Epilogue

The software has been mainly developed for learning purposes, though it gave a spark to an opportunity for those who wish to get down and dirty.

It should be noted that the game is protected by the **V**alve **A**nti-**C**heat, therefore tempering with the game's memory is a ToS breach.
The usage of the software entails taking full responsibility and accountability for any consequences.

UNIVERSITY OF PLYMOUTH

Vlad Burca
BSc (Hons) Computer Science
vlad.burca@students.plymouth.ac.uk

Languages

RE Tools