All rights reserved. No part of this book, GHFM- Game Hackers Field Manual, may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to the publisher using some social contact (all socials are Totally_Not_A_Haxxer). This book, GHFM- Game Hackers Field Manual, explores the realm of security-related programs and code. The content within these pages is intended for educational purposes only and should not be used to engage in illegal activities. The author and publisher do not endorse or promote any form of malicious intent, and the techniques described herein should be utilized solely for ethical and legal purposes. All names, characters, businesses, places, events, and incidents mentioned in this book are a work of fiction or used in a fictitious manner. Any resemblance to actual persons, living or dead, or actual events is purely coincidental. The unauthorized reproduction or distribution of this copyrighted work is prohibited. Criminal copyright infringement, including infringement without monetary gain, may be investigated and is punishable by law. The reader is solely responsible for the actions and consequences resulting from the use of the information provided in this book. The author and publisher disclaim any liability arising from the misuse or inappropriate application of the knowledge shared within. Please approach the content of this book with caution and adhere to ethical guidelines and legal frameworks in all aspects of your activities. Remember that knowledge should be used for the betterment of society and the enhancement of security and tech-related communities not to damage it. IMAGES OF GAME EXPLOITS, PROOFS, OR DESIGNS WERE ALL PULLED FROM GOOGLE AND ARE NOT REAL GAME CHEATS THEY ARE

FLAWED DESIGNS IN ORDER TO PROPERLY CONVEY AND EDUCATE THE AUDIENCE ON THE CONCEPTS AND IDEAS OF GAME EXPLOITS FOR ETHICAL REASONS. ALL CODE IS MADE UP AND IS NOT TAKEN FROM ANY DIRECT SOURCE ~ ***Do not sue pls and Don't be a fucking moron, thank you!***

0x0 Table Of Contents (TOC) 0x0

TOC: (p1-p16) | Book Introduction

TOC: (p16-33) | Intro To GH

TOC: (p28-29) | Getting Into It

TOC: (p29-31) | Getting Into It

TOC: (p34-38) | Terminology

Author's Note (Inserted During Editing): Hey there friend. I just wanted to say, before we fully start this book that I do fuck around ALOT in this book and I realize how much I do it. So understand that most of the content that is shown in this book is a bit scrappy, kind of fucking ass (I feel?), and will give you just a bare minimum introduction to game hacking. As I mentioned at the end of this book, there are many things you can look up on Google that this book could have talked about. And for that, I return my honor. Anyway, I would also like to address that I do shit talk ALOT in this book, mainly because the game hacking community is one of the most disgraceful communities anymore. Game hacking is an art form in my eyes, it's something that takes real skill and knowledge to do and it is something that can take people over a few years to master. There are many ways to do it- such as physical hardware and digital software and there are many techniques that are still being explored to this day. Of course, with this amazing part, you also have shitty SKIDS and LARPERS that come and trash artwork by stealing it and re-selling it on a market they can hardly understand, which makes me mad. I devote this book and every other future article and/or content piece related to game hacking, to the people who took to the skies in the 80s with arcade games and to modern game hackers who still try. I devote this book to those that are also willing to dive into the pool of knowledge filled with unknown opportunities and oh boy a FUCK ton of rage. To those who are starting game hacking as just a fun thing to do- then I certainly hope you are willing to dive as well. Go into the deep end with no experience and come back with a bunch of knowledge and experience. I also hope you become better than what the community is, better than a toxic dismemberment of society, and a taker of free will. This is your world- explore it, buy it, break it, build it, smoke it- do whatever you want but I say that in hopes you just stay away from trashing the parts of this world that make it stand out and make it unique.

I appreciate you buying this book- now enjoy :D

About Game Hackers Field Manual

I can not believe I am writing this section for the…what? 5th time? Anyway, let's do this shit again. If you are new here, welcome, to `Totally_Not_A_Haxxers` cave of wonder- really just a series of technical books and manuals that act as pocket guides or are manuals that I write for fun to either learn more about a topic or to spread my existing knowledge.

The Game Hackers Field Manual, Aka, GHFM, is a pocket-sized manual similar to **(BHGM, BHPM, VHFM, etc)** that is dedicated to highlighting key aspects of the game hacking realm which is what this book is dedicated towards. This kind of manual is supposed to be designed for people who just want some fun quick tips or knowledge in such realms- or are aiming to get back at their neighborhood paster or content-stealer who stole their game cheat (*cough*).

That being said, in this manual, we will go over many things but will aim to keep specific sections shorter than you may think- as I always say about ALL of my manuals- they are manuals, meant to be short, quick lookups that may be hard to find even with a few google dorks every now and then.

The main reason I decided to start this book- was one day, I made a post on one of my social media pages that revolved around game hacking, it was actually a contract by a small group of game cheaters asking me to help them in competition. When making this post, I got the drive again and remembered why I liked game hacking and what made me feel more like a part of the gaming scene. Thus, the inspiration to write a manual about it.

Its worth noting that I am not one to shake a tree and say my knowledge is 100% accurate and everything is 100% exact as it should be in fact, these books might not even be what you're (*ha, suck it grammar police, look at that perfect, moist “you’re”*) looking for. That is why I tell you honestly, as a weird fucking child looking to make a change and start with educating the general public with my experience.

Granted: this book might help out the wrong people, but we as developers, security researchers, and writers take that risk whenever we push a product out there. So I ask, if you are a kid looking to genuinely get into game hacking, this manual is for you- if you are looking to make money, copy a bunch of code, return this book- I feel painful that you are making this choice to even buy this book. In the end, I hope you do get what you are looking for because this book is for those looking to do something totally badass, not for the money, but because why the hell not!

What This Booklet Will Contain / Go Over

This book as you may have looked at the table of contents (for some reason, I see that people skip those- given the DMs I get) contains a weird set of information. Some of it is deeply informal, others going into basic exploit development and reverse engineering techniques. That being said, this book will be going over a few things.

Below is a bullet point set of the ‘brief’ contents of the book- if you need more context on sections, RTTOC (*Read The Table Of Contents*) PLEASE- other than that, good luck on inferring the contents based on section titles.

- **Introduction To Game Hacking As a Field**

- **Tools Of The Trade**
- **Choosing A Language For Game Exploit Development**
- **Windows and C++**
- **Design and Control aka Make Things Look Pretty**
- **Binary Security Overview**
- **Hacker Versus Hacker**
- **Resources, Forums, Using A Brain To Google**
- **Custom Tools With C++**
- **X86 Crash Course (not really)**
- **Scripting Game Cheats**

There are other primary sections, but based on this, you can infer A TON about this manual.

Warning [W]: Before getting into this booklet, note that when we do get into it, this book is not the content everyone imagines- it's not super technical, because it was not designed to be as other books and resources will cover various topics WAY better than this booklet can and I would prefer you go to some guy who has been doing this for 20 years vs some random child- but fuck it, your here.

Where Would You Use This Manual?

The purpose of this manual was not mainly to create something people can rush to for knowledge, as we have cleared that up the main purpose of it is to help give people access to a quick reference manual whenever they are stuck explaining concepts or want to practice specific concepts and understand them in an easier non-ELI5 (*Explain Like I am 5*) format.

So, let us build a sample use case. You are getting into the game cheat realm, starting your amazing journey but do not know where to start and do not know if you can trust someone's trashy C++11 source code to give you process information- this manual will showcase some scripts on getting process information and you can just wip this out of your pants and get started.

Now, in a deeper scenario. Say you got hired by some weird person on the internet to crack a small-time (*most likely pasted*) game cheat and is offering you money and you do not know basic RE concepts- then this manual can help you practice with that and you can learn from some of the concepts and ideas in this manual.

Or maybe you get lost one day and you want to learn more about how some people operate in this god-forsaken and terribly toxic environment, then you can flip to some page and either come across a very annoying personal rant about the community OR you can come across some information about what goes on in communities.

With that: this booklet is and has a wide variety of purposes- sometimes, it actually is not just about using the book- I actually expect some if not MOST people to buy this book and toss it up on a shelf with a million other books they forget to read or procrastinate on reading. But that is okay, because I wrote the book for hella fun and to see if this would be able to help me make a little and I do mean a little side cash to help fund my car hacking journey. We will get into the car hacking thing later.

It is also worth noting that this booklet is to add some form of personal satisfaction. I have read a TON of books from many different authors- especially on game hacking when I was mid-journey with the game hacking world. But I also noticed that a ton of the technical content was dry and was really helpful but not really enjoyable. When someone learns about game hacking, I picture them to be someone who does not know ANYTHING about tech but wants to do it for fun or to learn something new I can also picture some reverse engineers starting here (*such as myself and others I met along the way*). But no matter what it is, I want to put out information, give people a little laugh along the way, and make super boring topics fun to learn and super toxic environments a bit nicer to be around- game cheating is super boring sometimes and 90% of the time extremely toxic if you are in specific groups.

Here is something I want to end this section on- writing is hella fun, other times, really boring and you can burn out easily. I felt that just because I was bored I could release another book out of the MANY that I have in store- so, why not create more while I am at it? :D

Now, let's get into the goodies, shall we?

Totally_Not_A_Haxxer Who? What???

Before I got into this book, I wanted to give you a good background as to who I am and my experience before you go further into the book. You heard me mention *'car hacking'* like it was something I do for a living. If you guessed that, good job, or if you decided to take OSINT to another level and stalk me LOL.

Real start: My online alias is *Totally_Not_A_Haxxer*, this entire name is a meme and the logic behind it as to where it came from is brain-fucky- excuse my French. But, I grew up in the game cheating realm, it was my start, it was my main fun, and it was what I wanted to do and wanted to make a living with- that was until late 2022 when I fell in love with car hacking. When I fell in love with car hacking I was still in the field, but instead, I was building my own exploits and messing around in the game hacking realm to learn more, especially with open-sourced games. It wasn't until 2023 that I finally fully stopped game hacking and decided to just take it to another level and go to car hacking.

I explain this to everyone, the reason I left game hacking was primarily because it was not a career I saw myself doing forever after so long despite my love- most of it can get you into some shit, and prior experience just left me burnt out of the field. So, I spent some time finding a field that included exploit development, and reverse engineering, and also was new but not SUPER new. That field so happened to be car hacking. Ever since then, I spent my time messing more with reverse engineering firmware, writing exploits in my own programming language for fun, participating in friendly CTFs, and now practicing for certifications to fully finish my youngster road path. So, here we are. Retired game hacker, now moved to automotive cybersecurity, who wants to spread some knowledge on the world of game hacking in hopes of inspiring another child like myself to take a genuine interest.

Now, I do not want to get deep into myself- because this is not a book about me, and if you wanted to you can get enough about me quite quickly (*sadly*). So, shall we actually get into the book now? Sure.

Game Hackers Field Manual - Note Before Starting [Legal]

The license has already actually tried to emphasize this, but likely I will have to emphasize it over and over again.

DEPENDING ON THE GAME: Game hacking can be considered unethical in a shit ton of cases- but in all fairness, most companies toss a ton of bullshit in their EULAs such as ‘***misplacement of the game may be incorrect use of yada yada and breaks our EULA***’. Regardless, this book does not showcase game cheats or exploits for specific games, everything in this book is completely made up.

Endorsing the idea of building and selling game cheats is not considered unethical considering that this book does NOT target any form of audience, market, owner, group, company, corporation, etc, and does not tell people how to bypass, reverse engineer, exploit, or break specific systems belonging to any specific brand.

With that, I would also like to note that when we do showcase code, all code is meant to be conceptual, is not fully logical unless specified, and serves as a base to educate people on momentary topics.

I think it is also important that you (*the reader*) whether you are a lawyer or whatever the fuck you want to call yourself understand that no matter how hard you look or however you want to put it, this book simply gives people a good direction on how to start game cheating and where to go using games like **PwnAdventure** *(games built to be hacked)* as primary examples to start the game hacking path. So, please, do not ruin this fun time for me, or the readers. Not to mention, nothing illegal or unethical is happening here- literally nothing. Just conceptual understandings, algorithmic explanations, pseudo-code representations, and discussions about existing systems used to protect games. **~ Do not be a dick**

Game Hackers Field Manual - Structure

This section is dedicated to showing you some of the structure of this book and how this book will operate.

For people who are used to ***my*** books to be specific, most of this is not new to you, and you can go ahead and skip this section. For those who are not and do not understand my writing style, let me go ahead and toss this out there and suggest you definitely read this section.

- **Authors Note(s):** This is a field manual, but most of my field manuals are not the everyday 'flip through' kind of booklet. Instead, this book contains bricks of information in some sections and the complete opposite in others. Occasionally, you will come across some form of personal rant relating to the time writing this book or the topics we are talking about. Note that these sections are completely biased, are not informal, and are not designed to help you in the field this manual is dedicated to game hacking. But they can serve as some form of either laughter or confusion in your day. The reason I built my books this way is for the idea of creating something unique that most people are not used to based on the idea of unorganized writing. For example, in one book, I started ranting about space cats with burritos and then in the next rant, I complained about how horribly unstructured Python ***can be*** as a compiled programming language and then would also include more informal situations. None of these notes aim to be argumentative or angering but rather just absolutely random. Hence they are in this book- the sense of randomness, something machines were not designed for

- **BIAS Sections:** Sections in this book labeled with (BIAS) or [B] will showcase an opinionated section. These sections are in this book to formulate an opinion about a specific topic, technique, program, and so on from there. But no matter what, information in this book will not be labeled biased- as most of it is all technical and real-world applications but again, sections with *[B] and [R]* are readable but contain opinions about issues, sections with *[B] and [I]* are informational opinions about specific topics.

- **Information Note**: Informative notes are exactly that, notes left in the book are meant to be informative for specific sections or topics. Say we are talking about cracking software in the most basic form possible, I might make a note about how using standard methods of cracking software rarely works anymore- and how we may want to use other methods or I may also drop information on other existing methods used in specific environments to reverse engineer or crack software. Informational notes ***ARE NOT OPINIONS***, they are actual notes that can aid you in said topic.

That is pretty much all. But I would like to leave on one last tiny note. This book was written by a literal child- yes, that is who you bought a booklet from. A child with nothing but time on their hands to either work on their programming language or practice for certifications- needless to say, the reason I bring that up is not because of a comparison between age and knowledge, but rather a use case for inspiration.

All my life, people argued that it was not plausible to do something, it was not worth my time or effort to reach specific goals in life. But I am here, writing a book, at 16 my fourth book actually… (*most likely the third published but fourth written*), and still taking the time to do something because I wanted to do it.

If there's one thing to pick up from this book that is in any shape or form *'motivational'* it is that people will always try to make your path, pick your path, and try to steer you on that path to either control you or to mold you into someone they want to see- but its important to understand that you are in control of your own person. You can either let people tell you that something is not possible, or you can attempt it yourself, and make up your own mind.

If it is worth it in your own mind to do so- shoot for the ether- not the stars, the ether. Reach for goals that are beyond this world, reach for the ones that will make you, stay close to those who support you, and always make sure that what you are doing in life is making you happy. Fuck everyone who tries to turn that down.

Game Hacking - A Brief Introduction

If you are new to this kind of field, and bought this book to get some more insight, then this is the section for you. In this section we will go over some things in the game hacking realm, discuss what exactly it is, why it exists when it became a thing, and what this field entails. Also, it can prepare you for real-world cyber-security fields and even go into the current markets and state of communities right now.

So, without anything else to say- let's get into it!

Game hacking, yes, the fun side of the internet- where people who are complete nerds come and band together to build something that can fuck within the internals of systems. This is really interesting- but where did it all start? Well, to be completely honest- game hacking has been around almost the same amount of times games have been around- one of the most well-known starts in fact dates back the the 80s.

There was even a documentary called **'Console Wars'** that discussed individual companies that would modify the physical boards of gaming machines such as arcade machines just to make the game give players extra chances or individual advantages versus the *default(s)* on the games. All in all, game hacking has been around for a LONG time and definitely before multiplayer games existed.

But Why? Some people argue that game hacking in itself does not seem so fun. I mean, especially in today's games. Imagine this, you get home from work one day, tired as hell- and all you want to do is get some chicken wings and beer, and play ***Fortnite or COD (Call Of Duty)*** maybe some **MW (Modern Warfare)-** then the first game you hop on, some guy is flying around, auto killing every player in the game. Sounds boring does it not? Not to be the person being killed, but to be the person running the cheats?

Well, this is and is not true. As someone who has experience making game cheats and testing them, I will make the formal statement that it is quite boring- but at the same time, that is if you are using them in regular games. See, people do not always buy or make game cheats just to go cheat and up their scores- especially considering modern-day game companies not only have auto-banning systems to ban accounts like that but also considering the fact that most end up getting banned after what is called a ***manual reviewer*** (*a type of person hired by a game company to review gameplay footage in order to catch cheaters who manage to bypass AC [Anti Cheat] systems by spotting specific patterns in the gameplay log usually from a first-person POV [Point Of View]*).

Some people actually make and buy cheats just to taunt their favorite game cheaters. Imagine being that kid behind the screen or even an adult? Do you really want some joy out of your day? Go annoy your favorite most popular streamer in front of millions of viewers! Essentially taunting them haha!

That is what game cheating is about most of the time for the buyers. But what about the programmers? The people who spend the time out of their day trying to bypass AC systems and the people who spend well over weeks trying to figure out every single internal and external modification they can add to a cheat. To most, it's a money game, some cheats sell for thousands of dollars a month- so why not? But for others, it's an art form.

Manually reverse engineering a game, taking apart every single system, making sure you can slip some modifications in that are not super noticeable, and also predicting system calls and understanding how the program works simply by mapping out its logical calls- The point?

Game cheating is different for everyone, it is simply just another more unique form of hacking and computer hacking has many different meanings for everyone. But of course, that is the ‘why’ in the question.

For programmers- it is either art or super intuitive or it is a money game.

For users- it is either super fun to taunt people and ruin their day, or it is something they do (*depending on the game*) to boost their account status or cheat past competitions.

Fucking gnarly right?

An Outside Understanding For Newcomers

I am sure if you are reading this book and are new to this, you still do not quite understand the **‘art’** of game hacking. That is simply because of a few things- the main thing is the rage you feel when you get killed in a game because of a cheater or someone testing a cheat.

So, what exactly is the **‘art’** of game hacking? Well, you see, when you actually take the time out of your day as a game hacker to sit down and reverse engineer a game- it takes more than you know, and it takes a shit ton of knowledge just to put the pieces together.

Throughout the book we will go over some advanced concepts, we will also go over some basic ideas, algorithms used in games that hackers commonly use to build out cheats, and so on from there- but for now, let's get to know exactly how game cheats are built.

Let's take the most basic kind of cheat, ESP also known as Extra Sensory Perception. ESP is one of the most widely paid for and widely used cheats- this cheat pretty much in simple terms allows players to see more about their environment. Anything from the closest or farthest enemy to item information within the game. The following image may look quite familiar to you- this image showcases ESP an active game.

< UNLOADED ESP IMAGE FOR DEMO PURPOSES ONLY >

Seem familiar?

This kind of cheat for one takes a ton of time to make because think about it from the hacker's perspective.

In order to know how to draw those fancy boxes- or even start planning out the mathematics behind this- they need to do the following.

- **Find out how to pick apart the internals of a game**

Reverse engineering, as you have most likely figured out by now- is the process of taking something such as an application (*in this case, the game is the application*) and using a specific set of tools or framework to dump the internal information of the game to figure out how the game interacts with the environment, works with networks (*if it uses networks*), works internally, and so on from there.

But see, modern games are not just as easy as popping a framework open and dumping the game's information- now there are many security measures put in place while the game is running to prevent people from dumping or locating specific information within the game. That being said, they need to be able to properly bypass specific systems (*if it is truly needed*) or find ways to dump that information without triggering security systems.

When they get to this point, they need to go through all the information and pick out important information.

- **Programmatic Interaction**

When the game hacker or in this case, we will just say, reverse engineer is able to get all the required information such as what they need to do and the plan is laid out they now have to build a program that can interact with the games process which in very advanced scenarios will require bypassing security systems.

The reason for the interaction with the process externally using another program is because game cheats- in the most basic explanation are modifications that happen to the game as the game is running- and somewhere on the client, there is an application actively making those modifications to the game in memory. For example, a pretty basic game cheat will showcase the idea of reading and writing very specific values to the game's virtual memory - in specific locations of course.

The process interaction is not just about retrieving that same information the exploit itself needs, but also about injecting data into that process and there are millions of ways to do it.

You may think that based on my very shitty description of how a basic *(and I do mean BASIC)* game cheat works that there is still not a ton of work that needs to be done- but I also skipped at least 10 different steps.

Most of the time, just finding information can be a pain because sometimes you need to actually interact with the game using specific frameworks and tools (*dynamically*) in order to get some information out of it.

In summary: The point of this section is to at least give you again the smallest idea of how complex game cheats can be throughout this manual, you will see how deep this world really goes.

What Type of Game Cheats Are There?

Game hacking comes in many different forms- and requires many different bases of knowledge and existing skills to actually build cheats.

Break Point [i]: You may see me already saying *"actually build" but what do I mean by this?* In today's world, there is ChatGPT to help generate some script kiddies DoS script through very few sets of manipulation- well before ChatGPT, we could not generate code. Instead, game cheaters would rip apart existing code bases and cheats, dump their source code using disassemblers, and then either modify that code or copy and paste it and then resell. This kind of person is highly disrespected in the game-hacking realm and is quite hated. Simply because they take hours of work- no different than plagiarizing existing work or books and then re-sell it at either a higher or lower price. I have met a ton of these people in the time I spent here.

In order to understand what exactly game hacking entails, it is important that we go over some of the most common forms of game cheats.

Below is a bullet point-based list of types of game cheats and their descriptions.

- **Internal**: Internal game cheats are game cheats that usually fuck with the internal functionality of the game itself. I mean, to be honest, all game cheats mess with the game functionality- but internal game cheats typically differ in the way that they are used. For example, an internal game cheat may override internal functions and be injected via specific tools such as DLL Injectors.

- **External**: External game cheats are not injected into the game itself and require the game to be in a specific state (*sometimes*). A good example of an external game cheat is a game cheat that creates an overlay on the existing game screen to draw the skeletons or boxes around players that you saw in the last screenshot. Overlays are basically, invisible or transparent screens that are laid on top of the game window to draw and map locations of objects/entities on the overlay acting as if it is being drawn on the game.

- **Client-Side Cheats**: Client-side cheats are exactly how they sound. They are the most popular game cheat out there- because they are easier to build, cheaper, and do not require more knowledge of remote exploitation techniques.

Client-side game cheats simply just interact and work on the client's machine which means they typically do not affect any other player or online (*for instance*) entity within the same game.

- **Server-Side Cheats:** Server-side game cheats more or less have to deal with interacting with the game's server. For example, remember seeing those auto-nuke programs that would kill all players in the server at once when Fortnite first came out? That is a good example of a server-side game cheat. These can be much more sophisticated exploits (*sometimes*). Note that there are many different types of server-side cheats, the same with client-side cheats as well. We will go over this further in other lists or topics.

I know that was explained horribly (cough), but the difference is highlighted below.

Server-Side Cheats vs Client-Side Cheats | Summary

Client-side cheats operate on the client and only mess with the client side of the game- server-side cheats operate on the client-side but interact or exploit functionality on the game server.

Resuming: Just to clear this up, you may be assuming that game cheats can only happen on PCs, as most people do. The reason I am assuming this is because every time I get asked questions- it is always directed at how the hacker is operating on a PC.

Games can be hacked on any system, including web-based games, this also includes console games, hell, there is even a possibility of hacking NES games. Again, going back to the history of game hacking, this has existed long before consoles and long before home PCs existed.

Now, shall we get back to the second topic we were supposed to go over?

What Game Hacking Entails

Game hacking has many different fields included in it. This mainly depends on your goal, but, let's lay out a basic game cheat right now.

In the scenario that we are not copying and pasting code (*aka, actually doing the fucking work*) and we want to build out a small game cheat that can be sold and we want to make it a bit fancy with all those menus and stuff. So, let's go ahead and lay out the tasks that we need.

Below (*on the next page*) you will see a table of the skills that we need, followed by an ID of the bullet point that will discuss why we need that skillset.

Skillset	**ID of Point (IDoP)**
Reverse Engineering	1
UX & UI Development	2
Programming	3
Binary Analysis	4

Side Note: Our game cheat will be built in a multiplayer setting, an online game to be specific, we will call this- hot war (*made up game for the scenario*) that is pretty graphically intense, runs only on Windows 10, etc.

- **ID1:** The reason we need reverse engineering should not be anything new to any exploit developer or even existing hacker. The reason we need reverse engineering is that we need to figure out exactly what is happening throughout our game, how it works internally, where specific data is being held, how data is being stored, and more. This all comes in handy when we need to start writing the exploit. In the case that our game has AC (*Anti Cheat*) systems, then we need to find and reverse engineer the AC to find ways to bypass it.

- **ID2:** The reason for UX and UI development is for many reasons. Note that in this theoretical scenario, we are actively selling the game cheat to clients- we want to make sure that the cheat gives the user a good experience which requires a deep understanding of how to make the user experience work it and we also want to make a nice UI. Along with programming skillsets, we will need to understand UI design and how to make specific features just 'pop'

- **ID3**: Programming should be even more obvious than reverse engineering. There are many ways to make a game cheat, but if we are selling it and not building or automating via framework, then we will need to understand how to properly program the exploit and also how to manipulate processes, create fancy interfaces, or even basic user interfaces using that language, understand safe and direct development that can make the usage of the cheat software clean.

- **ID4:** Binary Analysis is another important skill because if we need to see what protection measures are thrown into the binary then we can usually use some viewers or frameworks to get information about the binary. For example, if the binary is stripped, uses or enables ASLR (*Address Space Layout Randomization*), or has the NX bit set, and so on from there. Most of this information does not require full-fledged binary analysis- because for god sake pwntools can get this info for you in seconds (*literally*) but it's worth it to at least know what you are looking at.

This is just a basic scenario and general layout to give you a simple idea of what game hacking actually entails. If you are building game cheats from scratch then it takes a ton of time- it's more than just taking someone's random code from forums and compiling it. It also may take knowledge of techniques as most of the time (*especially today given how good (kind of) AC systems have gotten*) you will need to employ very specific tactics and techniques to bypass protection systems.

With that, you can also expect some if not a ton of mathematics and general computer science knowledge with game cheating to be required. Granted: you can just have ChatGPT do it for you, ***but I highly advise AGAINST it considering that *most**** exploits you ever build might be quite complex and have large code bases, not to mention that GPT does not handle memory well O_O.

This should also give you an even better idea of the work it may take to build game exploits! See, it is not just a one, two, and three then BOOM you have a fully working game cheat. It takes time, it takes a ton of knowledge and expertise, maybe about a year's worth of constantly failing to perfect techniques, and a good amount of will to actually go through the steps of learning how to build one from the bottom up. Now I think it is worth getting into the **‘good people’** side of game hacking!

Hacking Games To Build Better Solutions

Whether you like it or not, game cheaters will always exist, they are always going to be there no matter how much you try to protect a game. In the same sense, cyber terrorists will always exist. This is simply because it is always hacker versus developer in this world of tech and not just that- because the money and the drive make people hungry to succeed in their mission no matter what the cost is.

So- why be a part of the problem? Well, game hacking is not just for people who want to make money or ruin someone's night- in the same way that hacking web applications or cars is not all for people who want to do harm. Instead, there is a good set of game hackers- people who are contracted to do good.

For example: there was a game company which will remain nameless for both legal and protective reasons, that had originally wanted me and some other person to help build game exploits for their game- this was because they wanted to understand how attackers bypassed their solutions and broke their servers to improve their security mechanisms. This, of course, had its challenges and the way the game was structured was weird in itself. But in the end, the point was clear- game hacking is not just for people who want to do harm, sometimes it's for people who want to help improve security solutions to ensure that game cheaters are less common.

Basically, game hacking can do many things for you, and you do not even have to target brand-specific games either- sometimes, you can be hacking games that were meant to be hacked like Pwn Adventure *(1, 2, 3)* and actually go through the steps of understanding how to fully uncover game cheat codes, vulnerabilities and modifications!

This can help you get better at not just protecting games- no no no my friend. It can get you better at just learning how to protect your own binary applications. Think about the amount of stuff game hackers have to do to modify applications and games (*which are just binary applications*) - learning how to do that yourself, can give you some good ideas on building some good solutions that can block many common forms of attacks. Will it be bulletproof? No, but it's worth the try!

Game Hacking - Getting Into The Toxic Realm

You may have heard me already refer to the game hacking realm as *'toxic'* simply because it is. But, I do not want that to steer you away from it.

This is because in reality, despite all the bullshit I have faced as well as others I know in this industry that are still active- if you drown out all the noise and take a passion for it- then the toxicity simply just won't exist. And even then, outside of the noise and the annoying 12-year-olds saying how secure their game cheats are and how "*uncrackable*" they are, it is a really fun field to get into no matter how you do it and it will prepare you for a ton.

Here are some tips I think are important as you go forward.

- **Cyber Hygiene Is Important / Opsec is important**

Cyber hygiene is just taking care of your own security. This includes using VPNs, making sure not many unknown or known know who you are, making sure that your username is not affiliated with any current usernames that are known, and making sure you are not boasting about your cheats.

- **Creating Real Software or Exploits**

A ton of people in this field copy and paste game cheat source code and re-sell it. The real connections in the game cheating world are found when you make real, quality programs and software *(both against exploit development and exploits themselves*). The community is helpful when you play nice.

- **Contributions Go A Long Way**

This is general advice for the world period, but making contributions to teams in the game hacking world (*ethical or unethical*) is extremely hard. So when you get the chance, if it fits your motive, then go for it. Because making those connections can get you access to a TON of knowledge.

Game Hacking - What To Expect When Diving In

The game hacking world is filled with so much information- I mean, so much. Information especially in the reverse engineering side has transferred over into my existing work and has helped me a long way.

But one of the biggest issues **(by far)** is how well hidden it is. Most companies when a game cheat is posted, usually will come in there and have the platform it was posted on remove it as a precaution to protect their software. This means that most educational videos on building or understanding game cheats pertaining to specific games or game companies will be quite hard to find ***(with good context*)**. When I say to find with good context, I mean a video that tells you exactly what is happening, what the code is doing, how the game reacts, how drawing on screens works, graphical manipulation, etc all works.

So, going in you can expect a lack of information, some people that do not want to help, and also not knowing where to look. In my eyes, it was a pain in the ass to find someone that even wanted to come close to educating me on game hacking or how it all worked- so I had to Google until I found books or resources, and then spent the time on my own understanding how it all worked by piecing it together.

You can also expect to be hit with a plethora of requirements to create even the most basic exploits. So I suggest you study at least intermediate to advanced of the following topics.

- **Computer Science:** To help you understand how computer memory works, operating systems work, programming languages, etc.

- **Exploit Development:** To understand how exploits work and are created

- **Reverse Engineering:** To understand how to pick apart games or applications and get information out of them.

- **Binary Analysis:** To understand how to analyze game binaries

Game Hacking - Breaking Into Digital Realms

This is the break point section of the manual- where we branch off into the manual itself.

I have already given you the smallest most Google-like introduction to game hacking, but anything after this section is where we get serious. You can think of this manual as one giant flipbook of my personal notes, the collision of articles, experiences, and stories that I have to share with you. So, as mentioned before, we will see some sections that are readable, others that are not that readable, and others that are simply terms and definitions (*such as the next section*).

I would also like to thank you (*the reader*) for taking the time to actually read or even use this book and even purchase it. These books/manuals/booklets or whatever the fuck you want to call it all started out as a genuine thing I liked to do- I was good at writing so I felt that I could do better writing books. My community has brought me quite a long way, pushed me to start projects like *BHRSM, BPHM, BHGM, VHFM, GHFM, Racing The Binary,* and etc. All of these books tied together are just the core representation of what I have wanted all my life- to find satisfaction in what I do to make money doing what I love and even be happy to share my knowledge and experiences with an audience no matter the size.

Regardless of the reason I am here, writing this to you- I am just happy to see that my work actually gets out there, to see it read by someone, to see it in the hands of people I looked up to, and to see people actually acknowledging that I have something I want to show the world or the groups I meet and that I have something to offer.

I hope this manual aids you in whatever task you are trying to accomplish or road path you want to go down- even if this information was horribly formatted *(as of editing I realize this)*, It may still be able to help you in some state or situation.

Best of luck, and I will see you throughout the rest of the book :D