



# The Search for a Title

A Profound Subtitle

Jaime Torres



Copyright © 2025 Jaime Torres

PUBLISHED BY PUBLISHER

BOOK-WEBSITE.COM

Licensed under the Creative Commons Attribution-NonCommercial 3.0 Unported License (the “License”). You may not use this file except in compliance with the License. You may obtain a copy of the License at <http://creativecommons.org/licenses/by-nc/3.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*First printing, March 2013*

# Contents

I	Part One	
1	Programs and Data .....	7
1.1	Programs and Data	7
2	C Pointers .....	9





# Part One

<b>1</b>	<b>Programs and Data</b> .....	<b>7</b>
1.1	Programs and Data	
<b>2</b>	<b>C Pointers</b> .....	<b>9</b>





# 1. Programs and Data

Before we know about local cybersecurity, we'll have to study the functioning and interaction of components in hosts. a **host** is in essence, any computer or similarly processing machine. As long as

Vulnerabilities can and **will** exist in these processing machines, such as

- Backdoors
- TOC/TOU attacks in shared processors.
- Memory operations

Vulnerabilities can be exploited, and when we

- Buffer Overflow attacks
- Malware
- Dumpster Diving attacks

We'll see them in more detail as we progress through this guide, so

## 1.1 Programs and Data

This guide assumes we're working on a Von Neumann machine, so all of our architecture will be assuming a memory, a processing unit, and interaction between both. Such instructions are binary in nature, and the existent framework we use has a few security implications, but for now we have to take them as a given.






## 2. C Pointers

```
int j,i, *p

int main()
{
    i = 5;
    i = j;
    i = v[j]
    i = *p
}
```





## Bibliography

