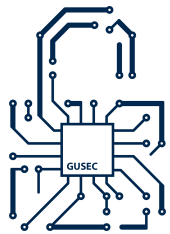
Reverse Engineering with Ghidra

Joe Rose

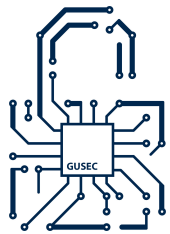
GUSEC

December 15, 2022



Who am I?

- 3rd Year Comp. Sci. Student
- Secretary of GUSEC
- Big nerd



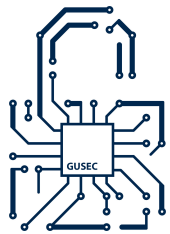
What is Ghidra?

Ghidra is a free and open source reverse engineering tool developed by the National Security Agency (NSA). Ghidra's existence was originally revealed to the public via WikiLeaks in March 2017, but the software itself remained unavailable until its declassification and official release two years later.

Ghidra is written in Java using the Swing framework for the GUI. The decompiler component is written in C++, and is therefore usable in a stand-alone form. Ghidra plug-ins can be developed in Java or Python

Supported Architectures

- Intel x86
- Intel x64
- ARM

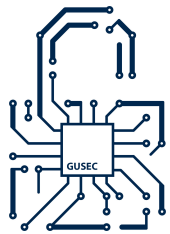


What is Reverse Engineering?

Reverse engineering, in the context of software, is the act of taking compiled code (also called a *binary*), converting that into a human-readable language, and figuring out what it does (also possibly modifying it).

What is it for?

- Static Malware Analysis
- Modifying closed-source code
- Creating closed-source compatible libraries



Why does it suck?

No Symbolic Information

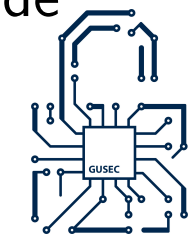
When we write code in a high-level language, we keep track of our data through variable names, we have no such luxury in reverse engineering.

No Type Information

Similary, high-level languages operate around the understanding of the well-defined data types and structures. At the binary level, we have to figure out the types ourselves

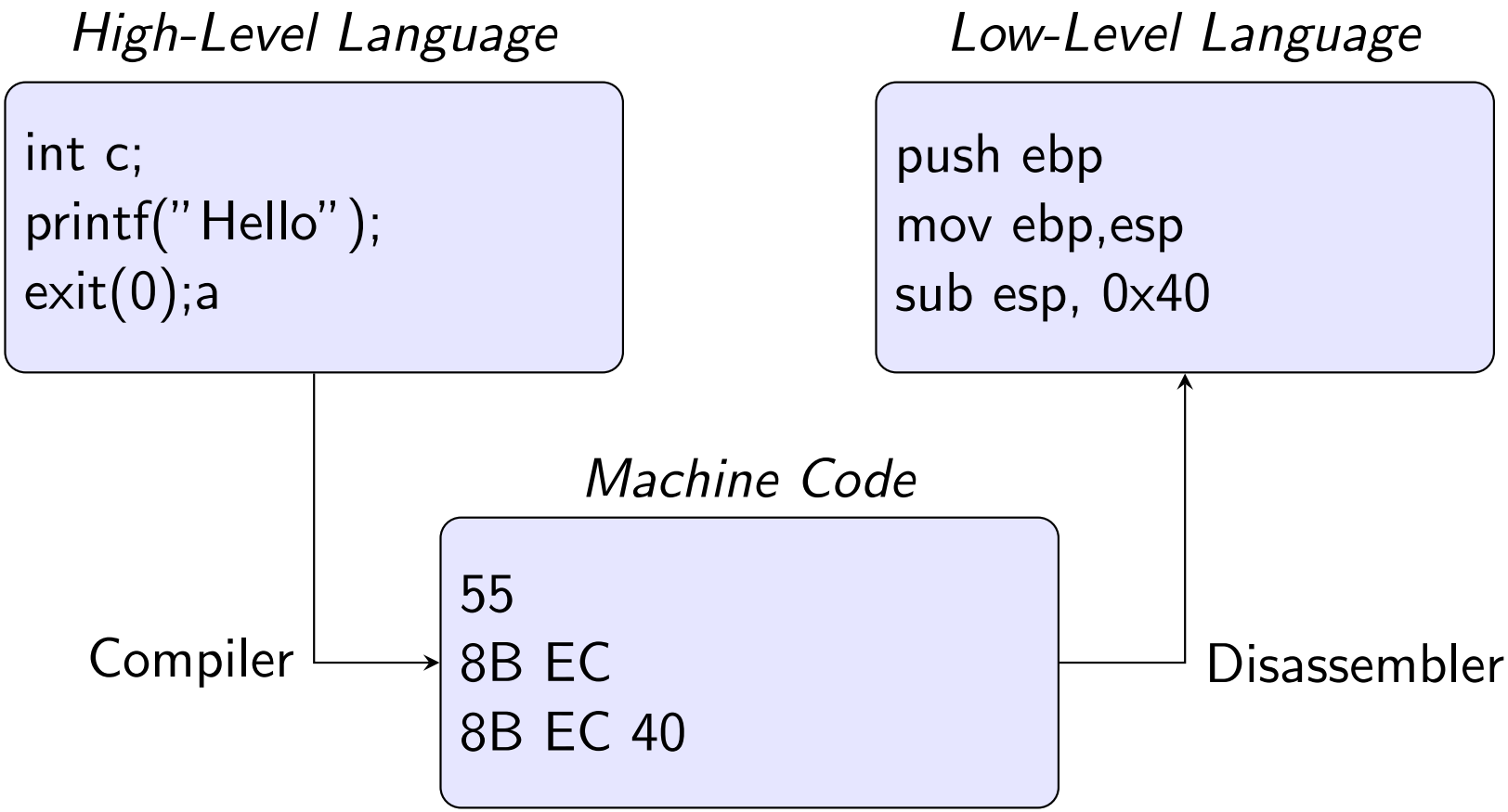
Mixed Code and Data

Binaries can and do contain data fragments within the executable code



Abstraction

Modern computer systems can be represented as layers of *abstraction*. As software engineers, we generally work in a 'high level' highly abstracted space, which is then converted by our compiler into low level *machine code*.



The Registers

Registers are extremely small (32-bit) pieces of memory within your CPU which are quicker than RAM or cache to access. These are what store the pieces of data currently being worked with. The x86 specifies some general purpose registers and some dedicated registers.

Register	**Purpose**
EAX	Accumulator
ECX	Counter in loops
ESI	Source in string & memory operations
EDI	Destination in string & memory operations
EBP	Stack base pointer
ESP	Stack Pointer