## Reverse engineering with NSA’s Ghidra

## Summary

## Reverse engineering with NSA’s Ghidra. Students using their own code will observe how different compiler options can impact the readability of the decompiled code. Students will decompile 3rd party code to determine an access password.

## Tools

Parrot: Ghidra, Ghidra is installed on Parrot by default. Downloads and help can be found here: <https://ghidra-sre.org/>

## Tasks

Copy the code from one of your previous ‘C’ assignments to a working directory on Parrot.  Optionally: ask ChatGPT to generate a bubble sort program for you.

* Compile the C code
* Start Ghidra and create new project
* Start the code browser and import the compiled program, yes to analyze code
* In the symbol tree window, select a function
* Compare the decompile code to your original code

Download a C program that requires a password (In Canvas Files: password.out)

* Determine the password by reviewing the decompiled code.

Download a Java class that requires a password (In Canvas Files: MyPassword.class)

* Determine the password by reviewing the decompiled code.

## Submission

Upload to Canvas (TXT, DOCX, or PDF file) with:

* A screenshot showing some of the decompiled function code
* Your original function code
* A description of what is different and why
* For both the password.out and MyPassword.class. Show string representation of the password the programs are looking for.

## Walkthrough

1. Copy the code from one of your previous ‘C’ assignments to a working directory on Parrot. Have ChatGPT write a bubble sort program if you don’t have any C program available.
2. Compile the C code

gcc bubble.c -o bubble.out

1. Start Ghidra and create new project, start code browser and import file to analyze

A screenshot of a computer

Description automatically generated

1. In the symbol tree window, select a function

A screenshot of a computer

Description automatically generated

1. Compare the decompile code to your original code
   1. Comments?
   2. Variable names?
   3. Function call changes?
   4. Storage of values?

Download a program that requires a password (In Canvas Files: password.out)

1. Determine the password by reviewing the decompiled code.

A screenshot of a computer program

Description automatically generated

The password string is an ASCII string stored a hexadecimal in reverse order. This is a compiler behavior, not all compilers use the same approach.

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

1. Repeat the analysis process with the file MyPassword.class

A screenshot of a computer program

Description automatically generated

## Additional Thoughts

Why is the password secret reversed? This behavior is compiler dependent. GCC pushes the characters on the stack a byte at a time, thus when popping them off the stack they are in correct order.

Does Ghidra support more than C and Java? Support is by CPU architecture. The processor list: X86 16/32/64, ARM/AARCH64, PowerPC 32/64/VLE, MIPS 16/32/64/micro, 68xxx, Java / DEX bytecode, PA-RISC, PIC 12/16/17/18/24, Sparc 32/64, CR16C, Z80, 6502, 8051, MSP430, AVR8, AVR32, and variants