Reverse Engineering

Mostly with Ghidra

Reverse Engineering Tools

- IDA Pro
  - Gold standard for 20 years
  - Expensive – requires licenses for each CPU you want to decompile
- Ghidra
  - Free and on-par with IDA Pro
- BinaryNinja
  - Built by CTF players
  - Highly scriptable with Python

Reverse Engineering Tools (cont'd)

- Radare2
  - Open-source community loves this tool
  - Painful for large projects
- Hopper Disassembler
  - OSX is the main platform
  - Good with ObjectiveC programs
- Objdump and other static disassemblers

Disassembling vs Decompiling

- Disassembling = Machine code to assembly
- Decompiling = Machine code to C code
  - Much harder task
  - Usually, the tool lifts the native code to an intermediate language and decompiles from there
  - Many different intermediate languages – every tool seems to use it’s own IL

Dynamic Analysis

- Linux
  - GDB
- Windows
  - Windbg – Native Microsoft
  - OllyDbg
    - Last release 8 years ago
    - https://www.ollydbg.de/
  - x64dbg
    - Actively maintained
    - https://x64dbg.com/
- Unicorn emulator – scriptable Python CPU emulator

RE Objectives

- Add enough context to understand the code
- Information is lost during compilation to native code
  - Compiler does not leave the data types of code
  - Compiler can re-use register variables as temporaries
  - Compiler can optimize and inline code

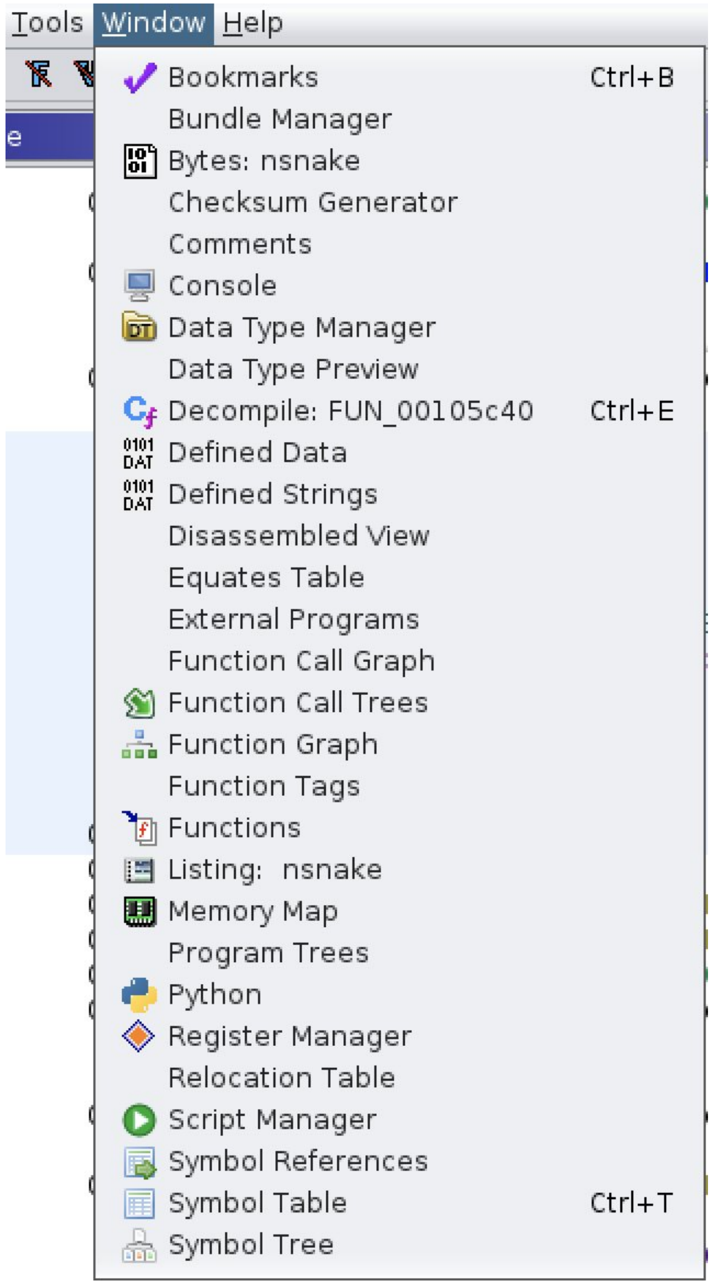
Main actions

- Follow the data
  - Identify data types and structure usage/layout
- Labeling
  - Provide a name to functions
  - Provide a name to function arguments
  - Provide a name to variables (stack, globals, heap)
- Add comments
- Use search and cross-references effectively
- Augment with dynamic analysis

Ghidra Windows

- Listing – aka Disassembly or dead listing
- Function Graph
- Decompiler

- Useful:
  - Defined Strings
  - Function Call Trees



Ghidra Window Rearrangement

- Drag the bar with the window title
- You can make split windows
- You can also make tabbed windows

Ghidra Debugger

- This is a separate tool
- You can connect to local or remote debugger
- Console can be weird on Linux when the program you are debugging wants user input

Next Steps

- Start reverse engineering CTF binaries
- Get the Ghidra book
- Go through the built-in Ghidra training
  - In the Ghidra zip file: docs/GhidraClass
  - 3 Classes in there