

Cyber@UC Meeting 104

// Reverse Engineering Basics //

If You're New!

- Join our Slack: cyberatuc.slack.com
- **SIGN IN!** (*Slackbot will post the link in slack*)
- Feel free to get involved with one of our committees:
Content Finance Public Affairs **Outreach** Recruitment Lab



Announcements / Upcoming Events

- We are currently 5th place in NSA Codebreaker
- 10/30: Rockwell Automation Visit / Demo
- 11/6 - Club Elections

Rockwell Automation Guest Speaker

Wednesday, October 30th - Rhodes 850D Patrick Feeley - Senior Embedded Software Engineer

- An overview of Industrial Control Systems (ICS) and how they relate to cybersecurity
- History of ICS Security (Stuxnet, Ukrainian Power Grid Attacks, Trisys, etc.)
- Challenges in ICS security attacks
- Potentially a DEMO of how security issues in ICS can be exploited to cause physical damage

Bring Resumes and Questions!

Reach out to @Michael Sengelmann on cyberatuc.slack.com if you have any questions



+

Rockwell
Automation

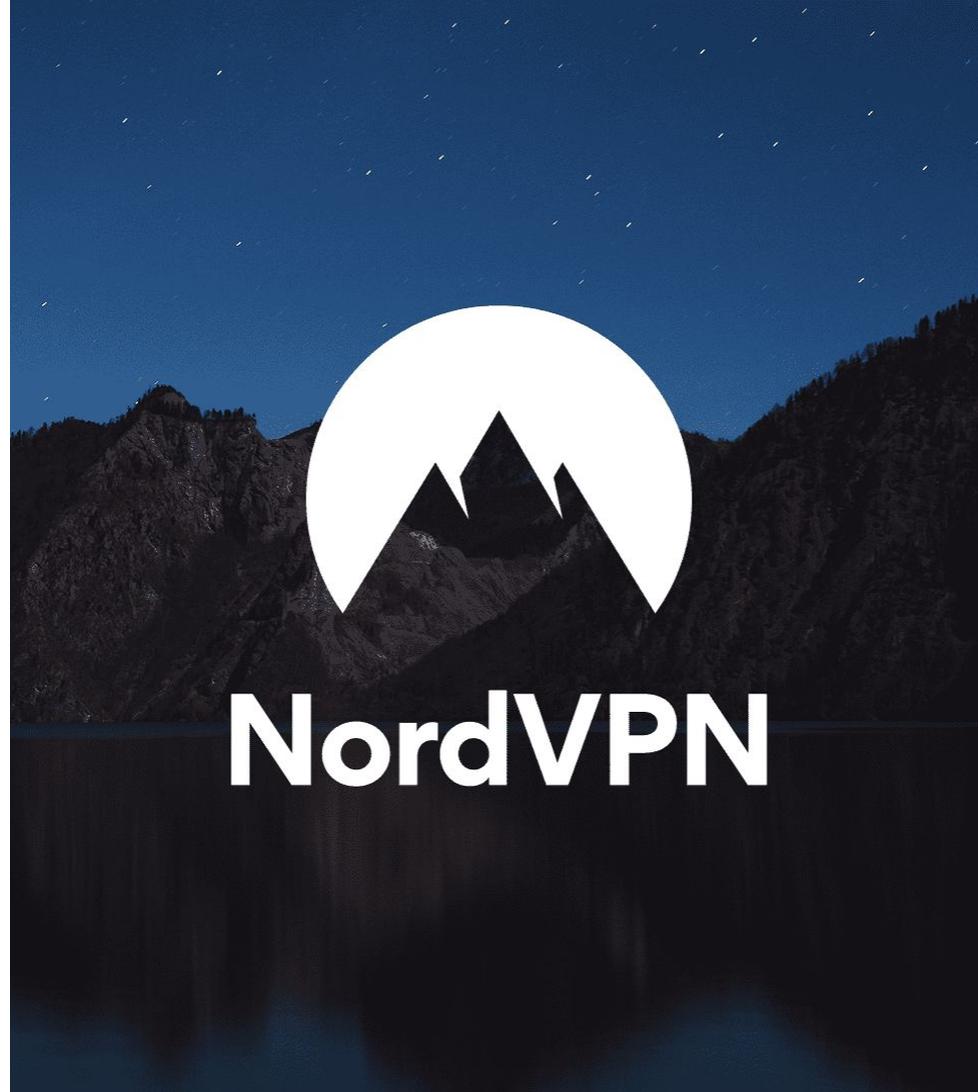


Weekly News

NordVPN Breach

- Insecure remote management system account
- Attacker could spin up their own server
- 1 of 3000 servers
- Would need the client to ignore the expired key

<https://nordvpn.com/blog/official-response-datacenter-breach>





Reverse Engineering Basics

Agenda

- I wasn't here last week!
- What is REeee
- Why REeee
- REeee Tools
- Binary Compilation Process
- CTF Challenge from Battelle
 - Goats walkthrough w/ ghidra

I wasn't here last week!

- Shame on you
- Jason Armstrong from the NSA came and gave us an incredible talk on the history of Encryption and even brought an original enigma machine for us to play with



What is Reverse Engineering?

- Process of analyzing software to figure out how it works, how it was written, and more
- Typically done with a combination of debuggers, disassemblers, and decompilers
- Static analysis
 - Inspect the program without running it
 - View code, draw conclusions
- Dynamic analysis
 - Inspect the program as it runs

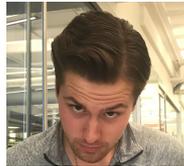
Why Reverse Engineer Things?

- Figure out how things work
 - Change how things work by extending them
 - Find vulnerabilities
- Reverse engineering is used to:
- Make exploits
 - Hack video games
 - Win CTF's (like CodeBreaker)



Reverse Engineering Tools

Binary Tools (ELF / PE / MachO)	Android / Java Tools
GHIDRA diStorm3 IDA edb-debugger OllyDbg Valgrind YARA Strings R2 / Cutter Binary Ninja	GHIDRA apktool dex2jar jad jvasnoop jd-gui smali

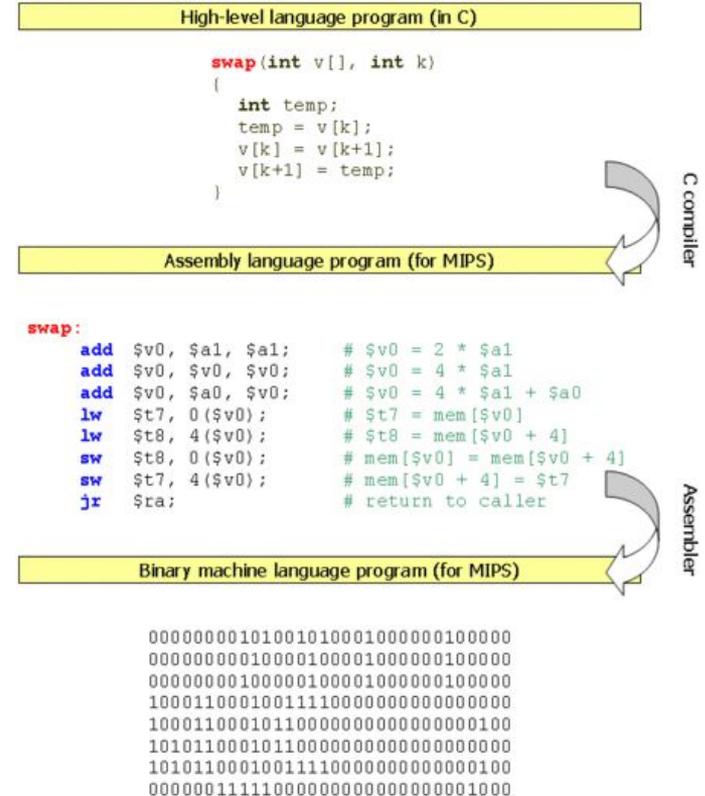




GHIORRA

Compilation Process - Executables

- Source code is written in language of choice (here in C)
- Code compiles to assembly / interpreter code
- Native Code (C++, C, Rust) continue being compiled to the actual numbers that the processor runs
- On some scripting languages, compilation may not be done at all as the script is interpreted by the language binary

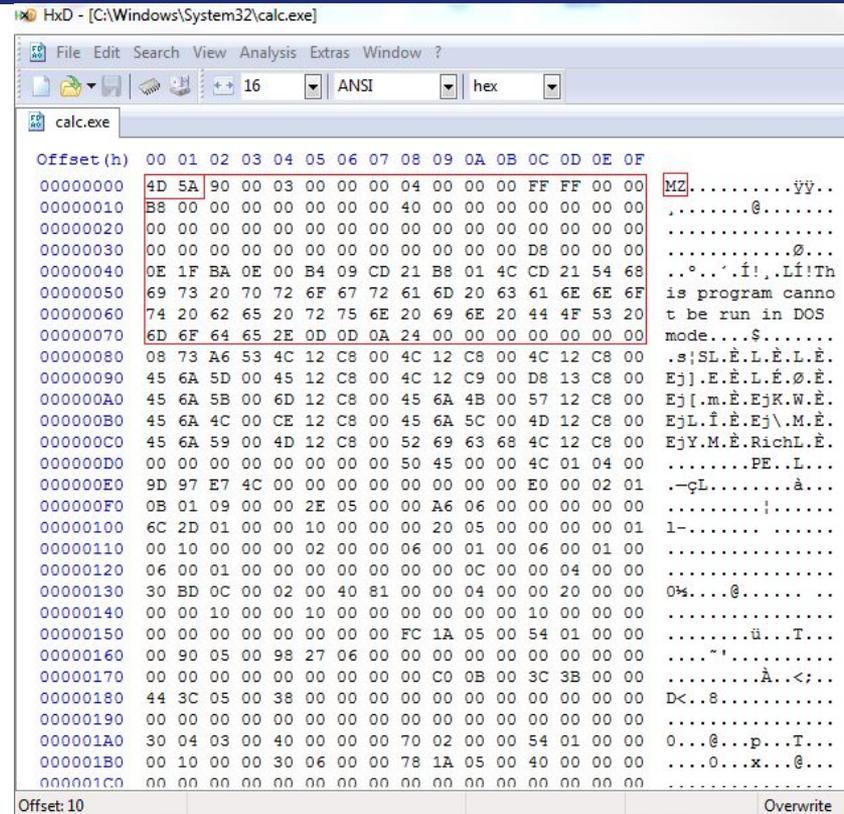


Compilation Process - Dependencies

- **Static Linking**
 - Dependencies are included in your output binary (.LIB/.O)
 - Pros: much more portable, single output program
 - Con: larger output binary size
- **Dynamic Linking**
 - Dependencies are looked up by the OS when the program is run (.DLL/.SO)
 - Pro: smaller individual binary size, multiple programs can share deps
 - Cons: Dependencies might not be on machine, more files to track
- **Run-Time Linking**
 - Like dynamic linking except the program finds the dependencies it wants to load manually and then pulls them into memory
 - Pro: Reverse engineers take slightly longer to see your dependencies
 - Con: Almost exclusively used by malware, so if you're trying to make a video game hard to hack you'll probably be blocked by antivirus from even installing

Compilation Process - Strings

- Typically the strings in your application get shoved into a special section of the binary in ASCII or UTF-16 format
- All modern Windows program include “This program cannot be run in DOS mode” right at the beginning as well
- We can scan a binary in a matter of nanoseconds



```
Offset (h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000 4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 MZ.....ÿÿ..
00000010 B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 .....@.....
00000020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000040 0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68 ...°.!.Li!Th
00000050 69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F is program canno
00000060 74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20 t be run in DOS
00000070 6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00 mode.....$.....
00000080 08 73 A6 53 4C 12 C8 00 4C 12 C8 00 4C 12 C8 00 .s!SL.È.L.È.L.È.
00000090 45 6A 5D 00 45 12 C8 00 4C 12 C9 00 D8 13 C8 00 Ejj.E.È.L.È.Ø.È.
000000A0 45 6A 5B 00 6D 12 C8 00 45 6A 4B 00 57 12 C8 00 Ej[m.È.EjK.W.È.
000000B0 45 6A 4C 00 CE 12 C8 00 45 6A 5C 00 4D 12 C8 00 EjL.î.È.Ej\M.È.
000000C0 45 6A 59 00 4D 12 C8 00 52 69 63 68 4C 12 C8 00 EjY.M.È.RichL.È.
000000D0 00 00 00 00 00 00 00 00 50 45 00 00 4C 01 04 00 .....PE.L...
000000E0 9D 97 E7 4C 00 00 00 00 00 00 00 00 00 00 02 01 .-çL.....à...
000000F0 0B 01 09 00 00 2E 05 00 00 A6 06 00 00 00 00 00 .....!.....
00000100 6C 2D 01 00 00 10 00 00 20 05 00 00 00 00 00 01 l-.....
00000110 00 10 00 00 00 02 00 00 06 00 01 00 06 00 01 00 .....
00000120 06 00 01 00 00 00 00 00 00 00 0C 00 00 04 00 00 .....
00000130 30 BD 0C 00 02 00 40 81 00 00 04 00 00 20 00 00 0%.....@.....
00000140 00 00 10 00 00 10 00 00 00 00 00 00 10 00 00 00 .....
00000150 00 00 00 00 00 00 00 00 FC 1A 05 00 54 01 00 00 .....ü...T...
00000160 00 90 05 00 98 27 06 00 00 00 00 00 00 00 00 00 .....!.....
00000170 00 00 00 00 00 00 00 00 C0 0B 00 3C 3B 00 00 .....À.<:;...
00000180 44 3C 05 00 38 00 00 00 00 00 00 00 00 00 00 00 D<..È.....
00000190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000001A0 30 04 03 00 40 00 00 00 70 02 00 00 54 01 00 00 0...@...p...T...
000001B0 00 10 00 00 30 06 00 00 78 1A 05 00 40 00 00 00 .....0...x...@...
000001C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```



BATTELLE