



INTRODUCTION TO

{ XenoboxX }



Hardware Sandbox Toolkit

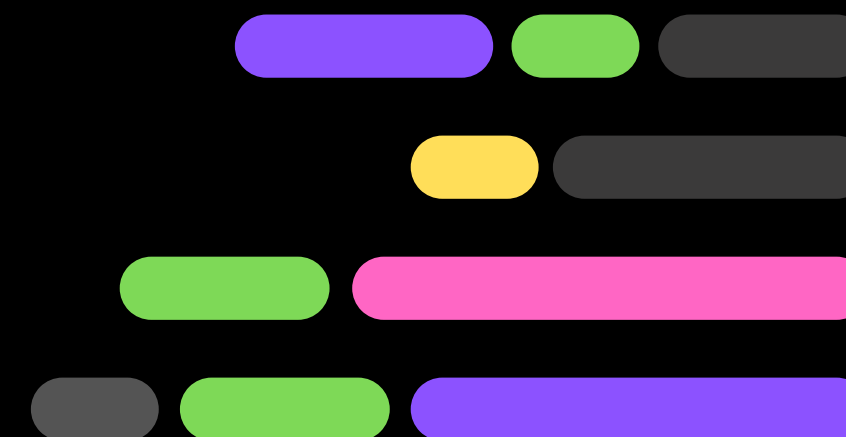
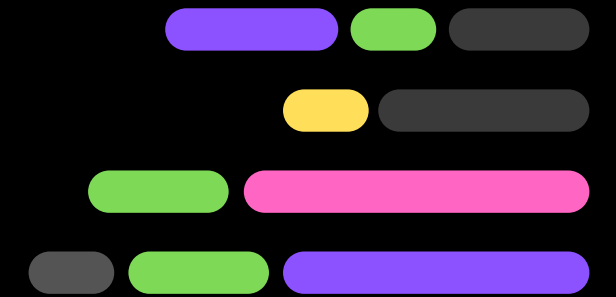




TABLE OF CONTENTS



01

WHO

Introduce myself

02

WHAT

A brief description

03

HOW

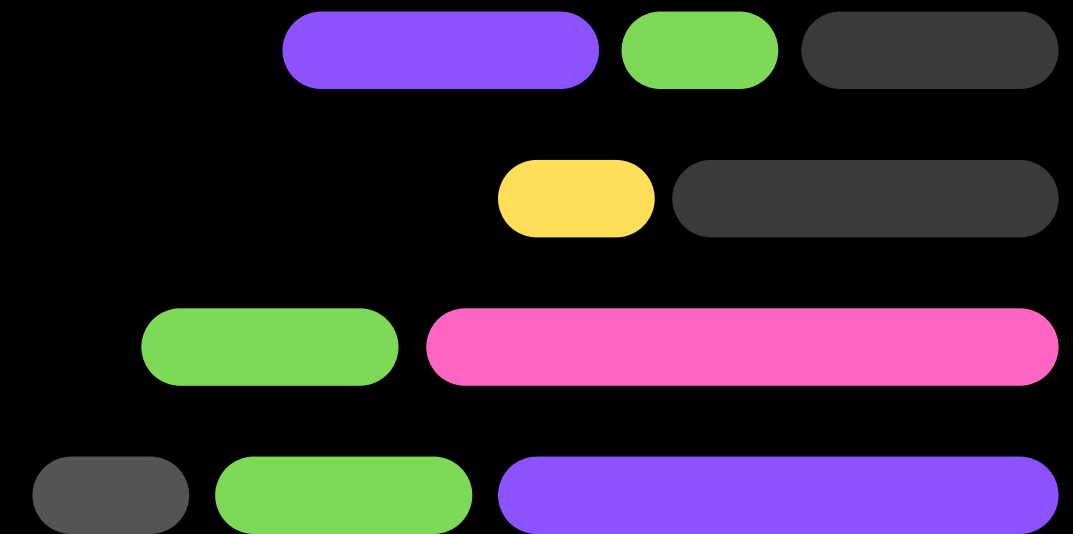
DEMO time

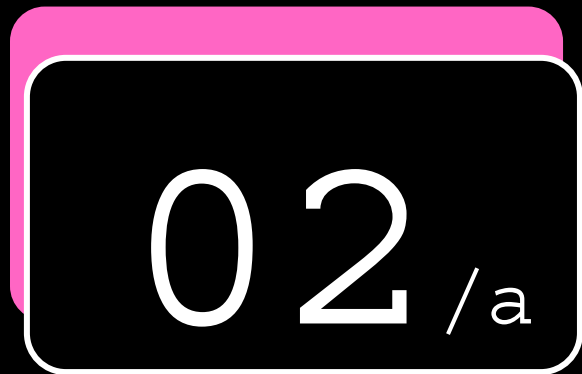


WHO

Cesare Pizzi

- Security Researcher (to pay bills)
- Security Researcher (as hobby)
- OSS Dev: REW-sploit, USBvalve, SYNwall (plus contributions: volatility, TinyTracer, etc)

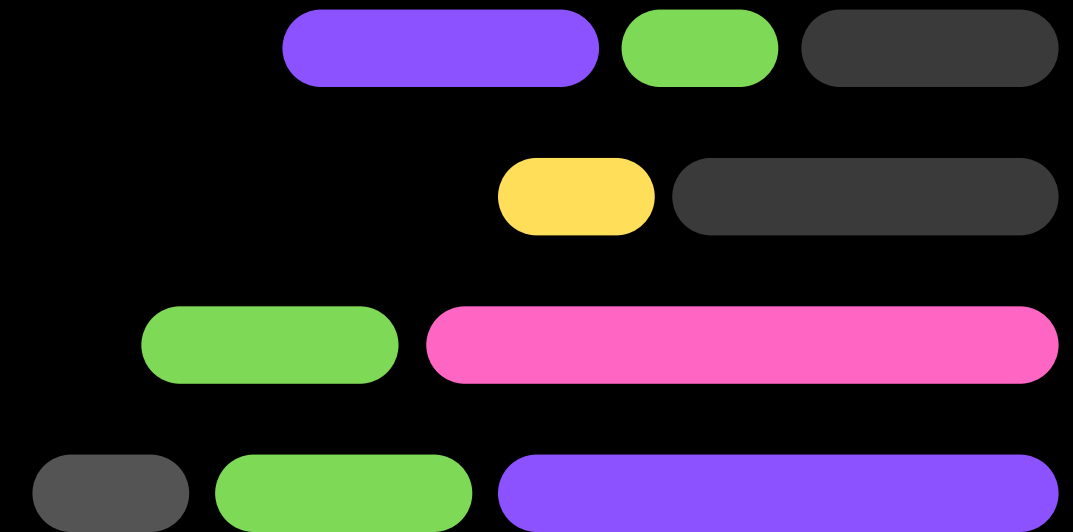




WHAT

XenoboxX:

Malware frequently employs anti-VM techniques, which can vary in their difficulty to detect and counteract. While integrating anti-detection measures in our labs is a frequently used option, we should also consider using a real hardware sandbox, even if this sounds weird.

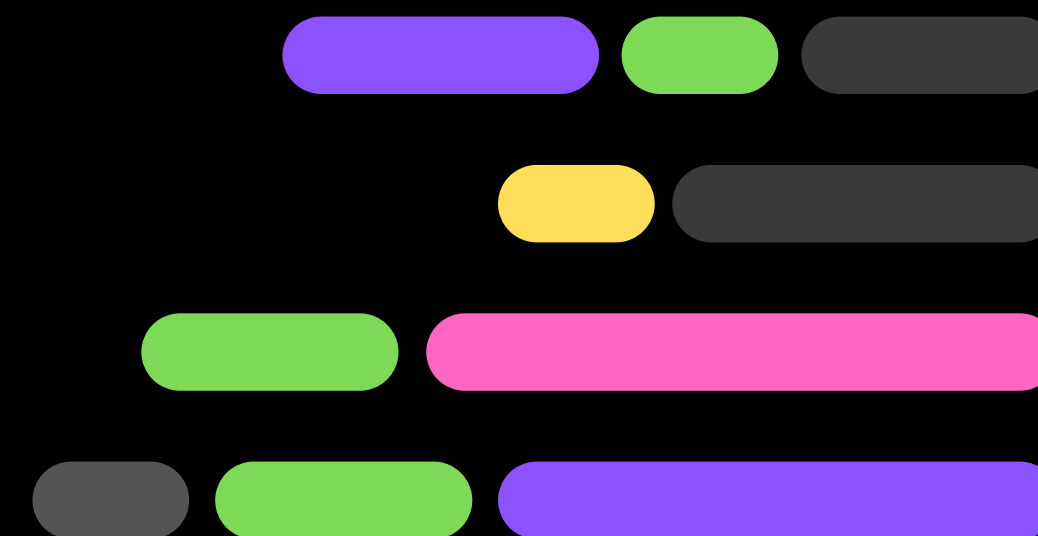


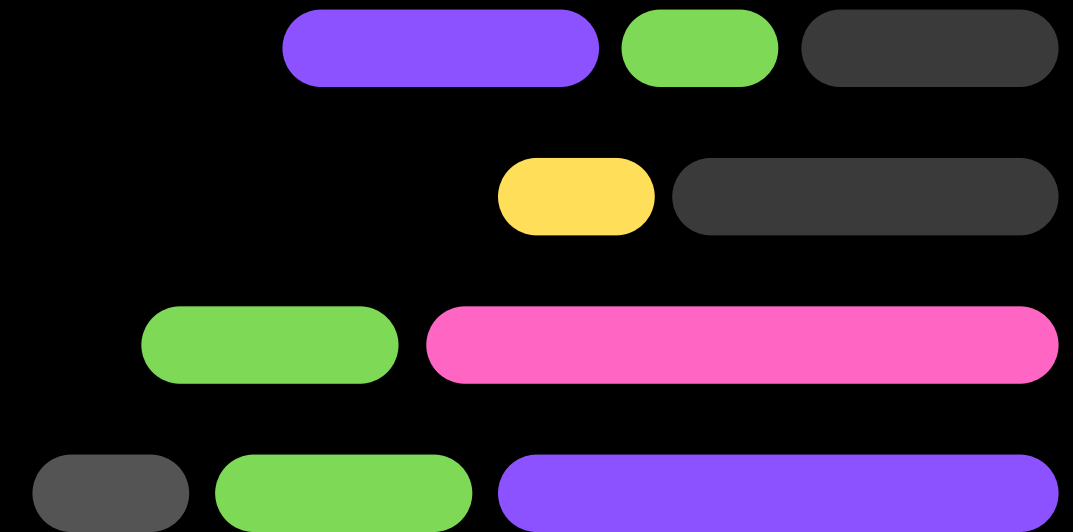
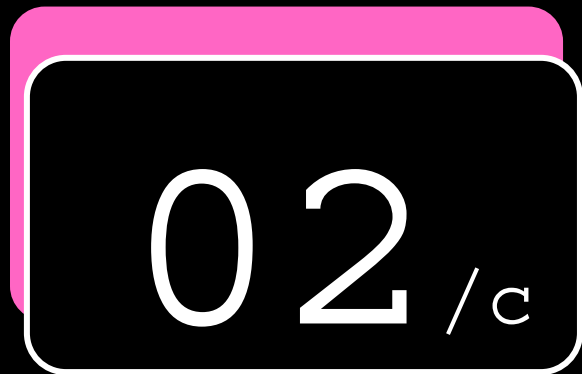


WHAT

XenoboxX:

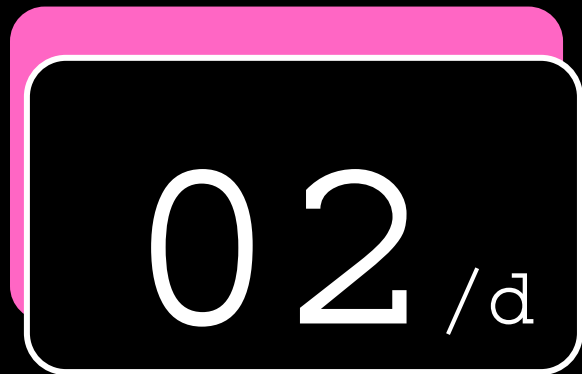
By leveraging the awesome [PCILeech](#) project and DMA hardware access, **XenoboxX** provides a suite of tools for analysis tasks, such as dumping dynamically allocated memory and searching for IoC. These tools allow us to inject code at kernel level through DMA, making detection significantly more challenging and giving a new perspective to the analysis.





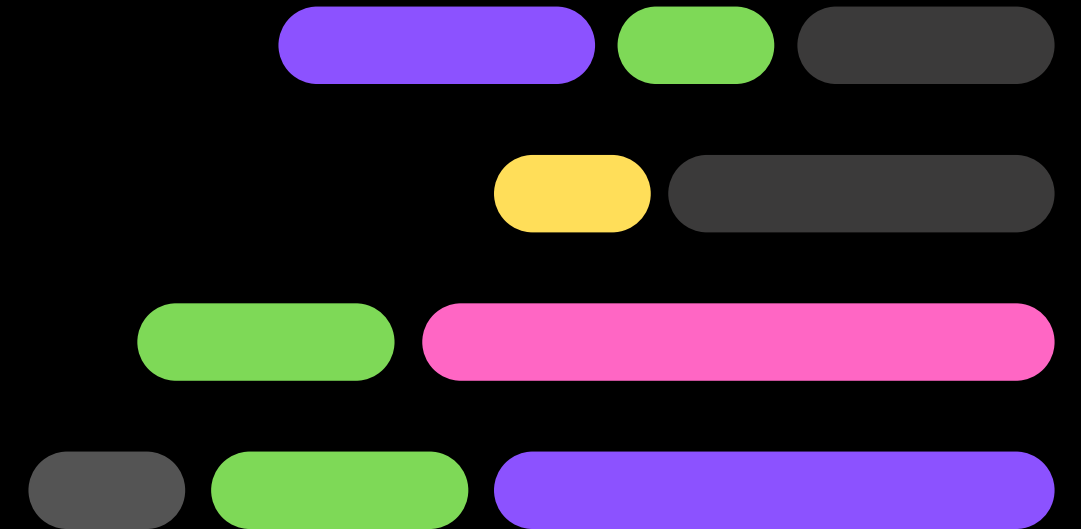
WARNINGS

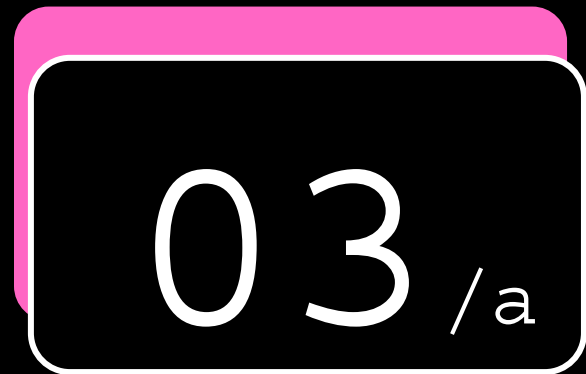
- XenoboxX is currently focused on 64 bits Windows, but in the future it could be extended for other platforms too.
- As an experimental approach, Xenobox currently avoid hooking at all, and all the detections are done through polling. Obviously this has some drawbacks, but this is a design choice at the moment, in order to be as stealthy as possible.



WARNINGS

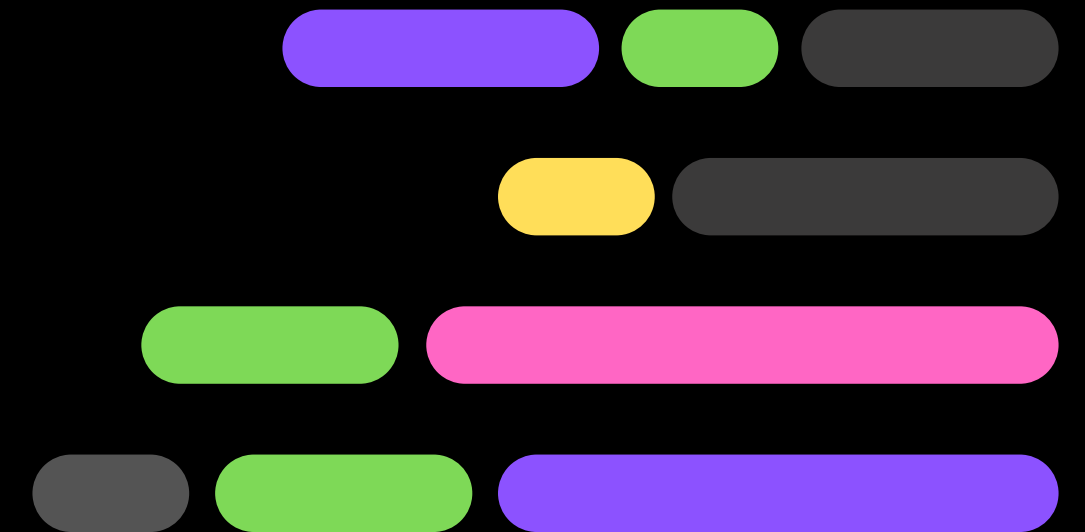
XenoboxX is not the definitive analysis tool: this is a very **specific approach** I find useful for very specific analysis: I don't think this tool is going to replace your current workflow, but maybe you'll find it useful for that specific **nasty** malware

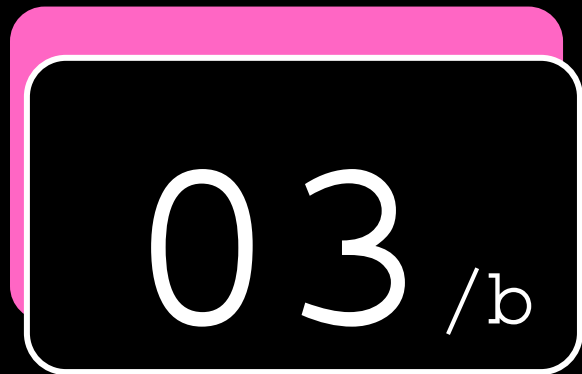




HOW

- **XenoboxX** leverages the entire environment of **PCILeech**, using its Kernel module to inject shellcodes.
- **PCILeech** provides some custom shellcodes for different OS (wx64, Linux and Mac)

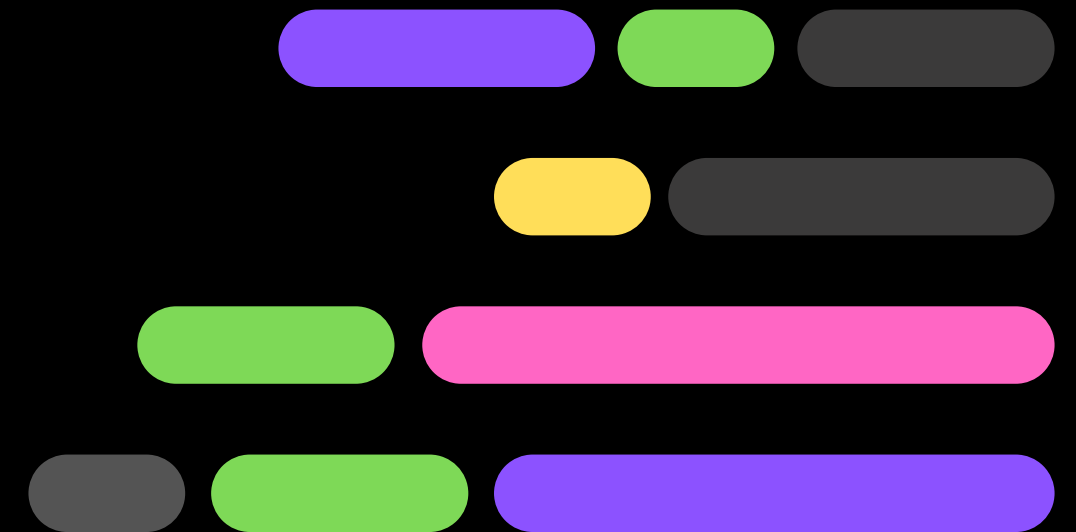


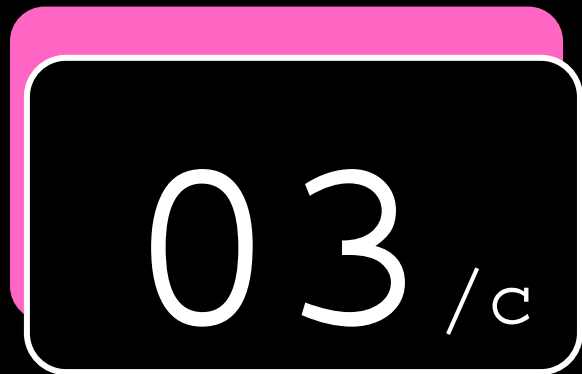


HOW

- **XenoboxX** currently provides 3 custom shellcodes:
 - wx64_dumpalloc
 - wx64_memgrep
 - wx64_strings

All are injected in kernel space and can attach to any process on the system.

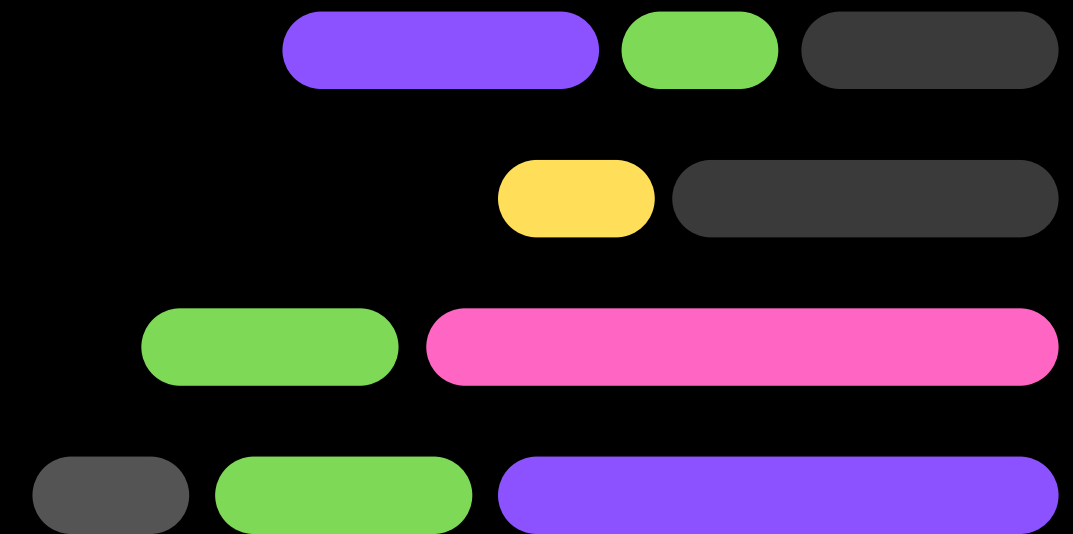




HOW

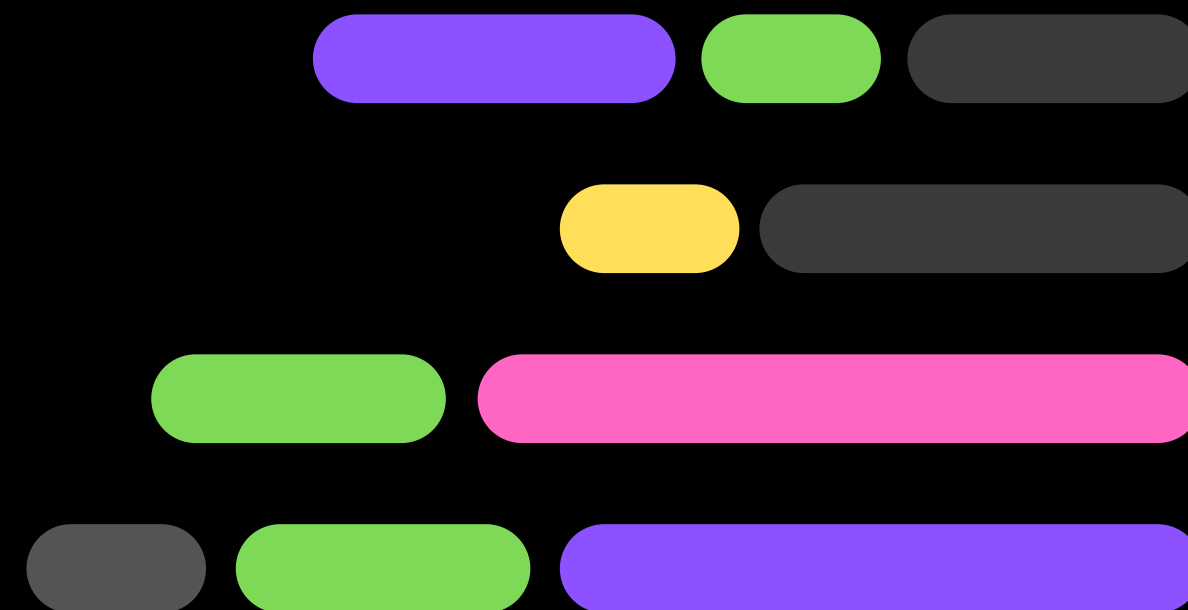
What you need:


- target PC with PCI interface, with a “quick restore” system
- DMA Board
- PCILeech
- XenoboxX





DEMO Time





THANK YOU!

Where to find XenoboxX:

<https://github.com/cecio/XenoboxX>

Special thanks to PCILeech/Ufrisk:

<https://github.com/ufrisk/pcileech>

Slides template from:

<https://www.slidescarnival.com/>

