Memory Forensics and DMA Attacks with MemProcFS and PCILeech

2019-10-23

UlfFrisk

MemProcFS: Fast easy-to-use Memory Analysis

PCILeech: Direct Memory Access (DMA) Attack Software

Agenda

DMA-attacks with **PCILeech**

- **PWN LINUX** and **WINDOWS** with DMA code injection
- **MOUNT** live file systems
- Spawn System **Shell**

Memory Forensics with **MemProcFS**

- What is **The Memory Process File System?**
- In-Depth: Capabilities Design, API and Plugins

Demos - **Live Demos**!

whoami: Ulf Frisk

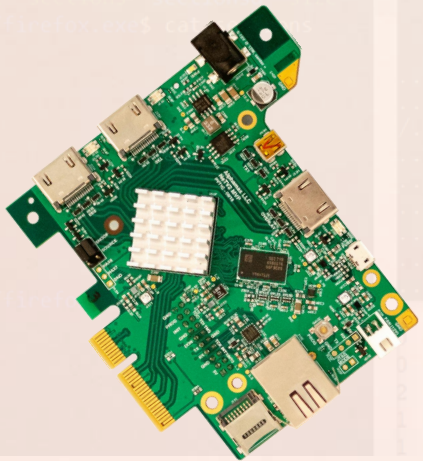
Pentester by day @ Polisen IT – Stockholm, Sweden

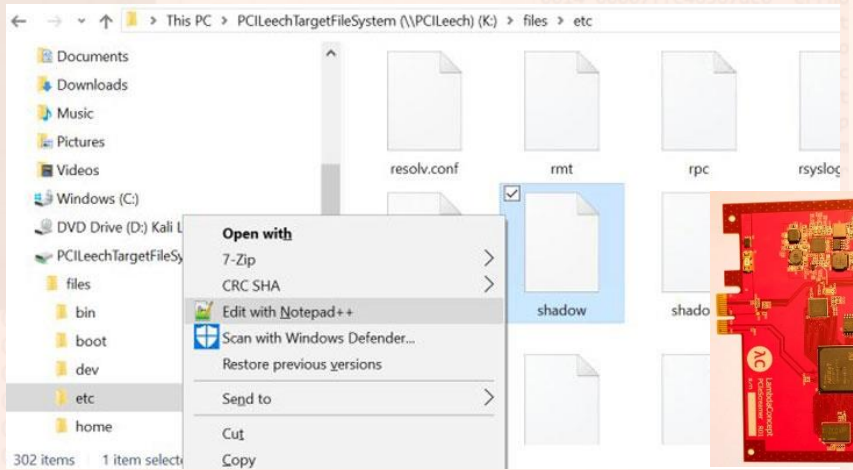
Security Researcher by night

Author of the PCILeech Direct Memory Acccess Attack Toolkit

Presented at different cons including DEF CON, BlueHat and the CCC

100% Open Source



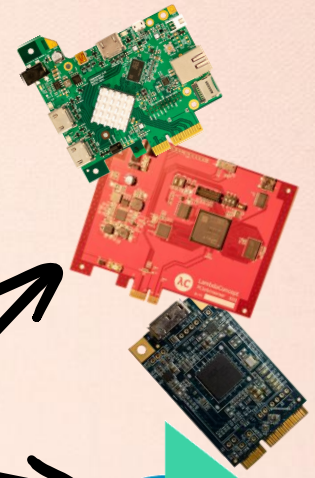


PCILeech and MemProcFS

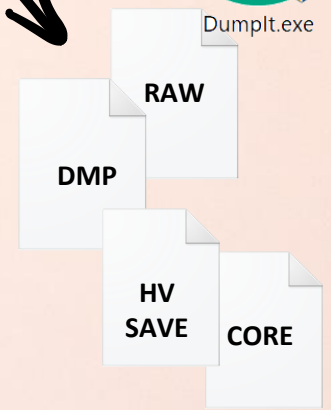
PCILeech

MemProcFS

LeechCore Library



Dumplt.exe



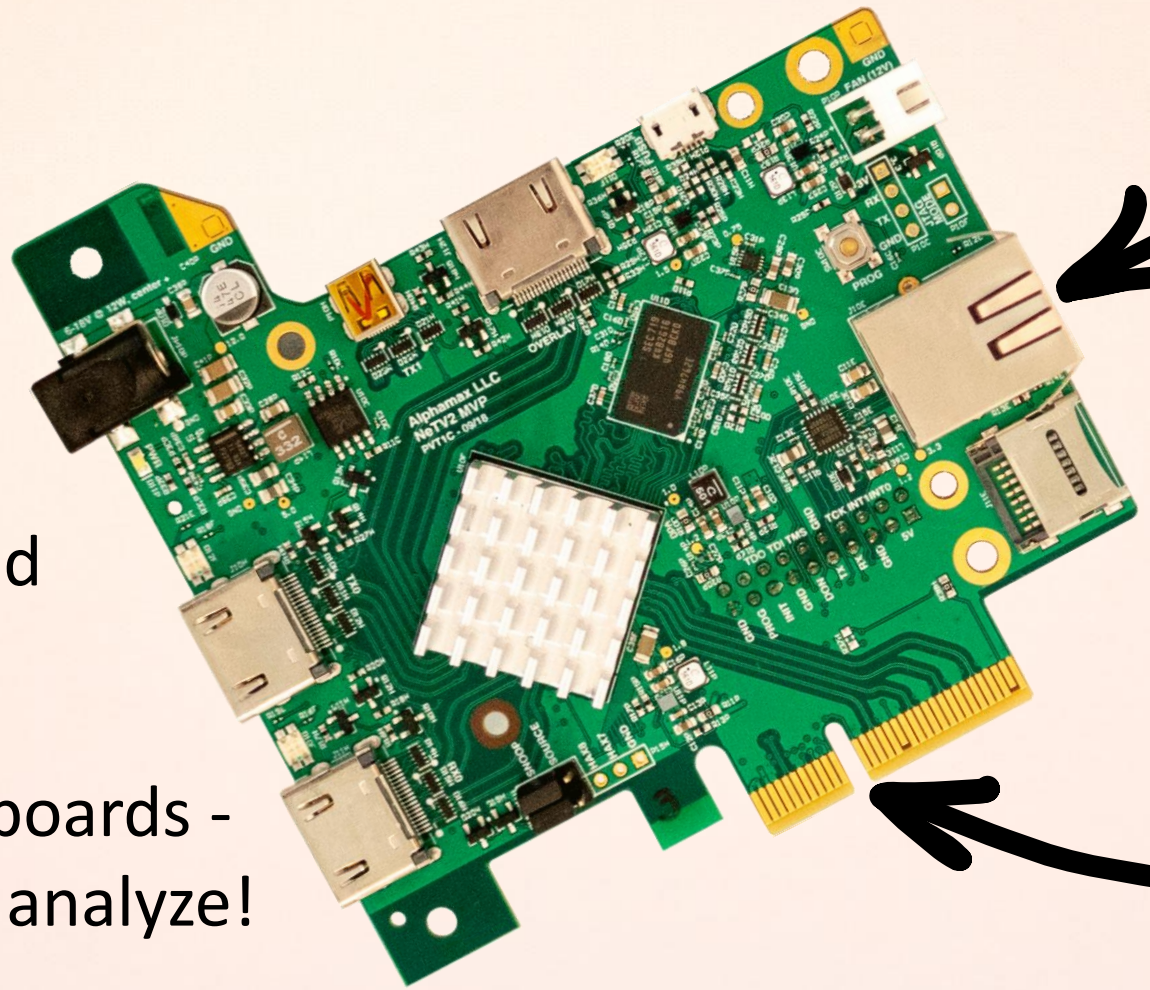
PCILeech

A New Hardware: The NeTV2

github.com/ufrisk/PCILeech

The NeTV2

100Mbit ETH
~8MB/s DMA



$215

Cheapest Supported
FPGA Hardware

slower than other boards -
plenty for attack & analyze!

PCIe

PCILeech

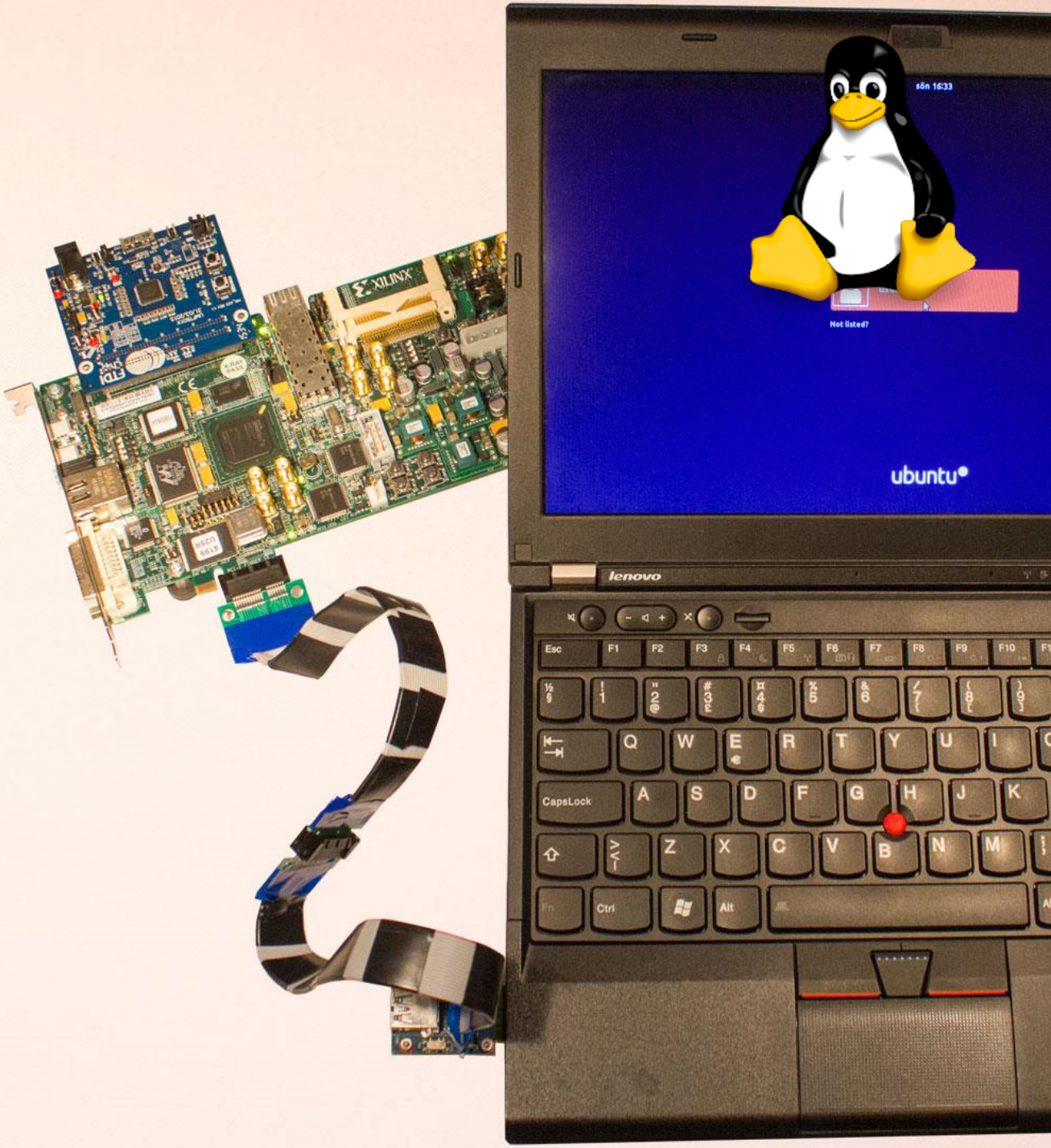
DEMO: LINUX

github.com/ufrisk/PCILeech

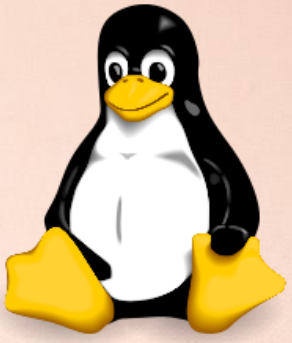
Linux DEMO

KERNEL IMPLANT
MOUNT file system
UNLOCK

Target System: Kali 2019.3



Linux Security



Hardware without DMA ports

BIOS password, DMA port lockdown, Pre-boot authentication

IOMMU / VT-d (virtualization-based security)

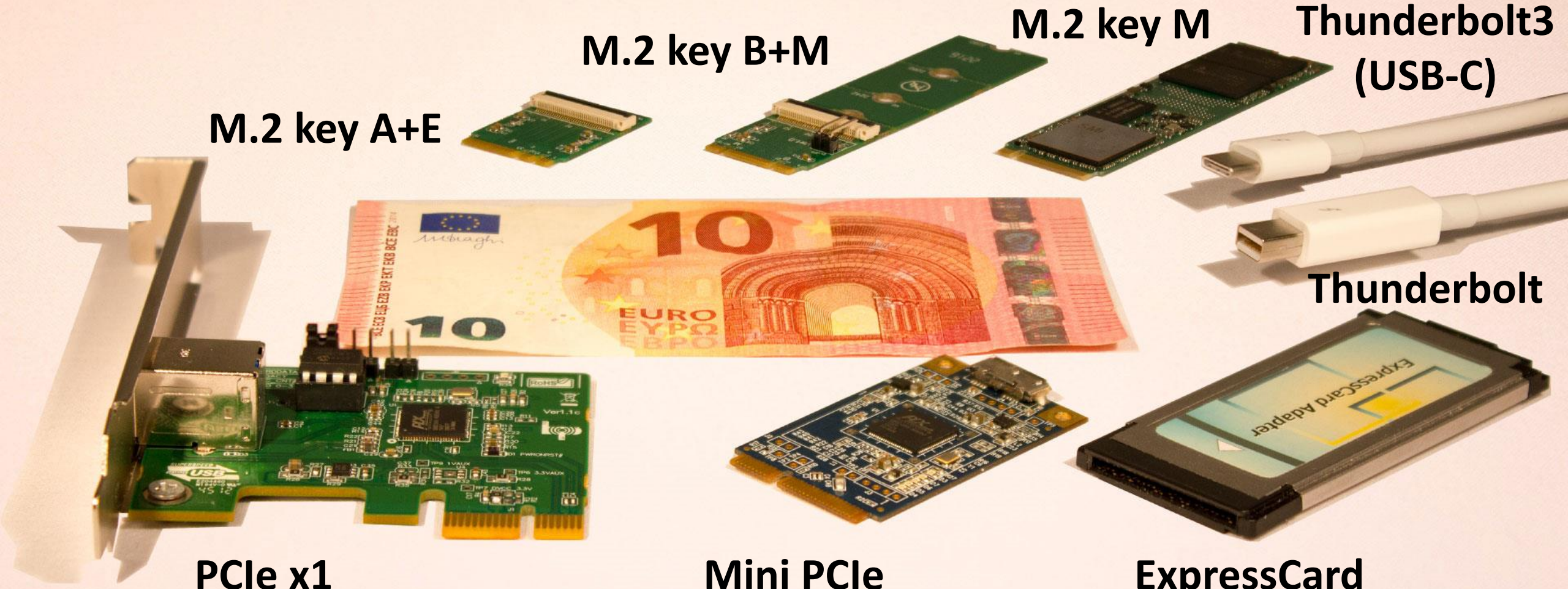
 **== secure by default!**

PCILeech

PCI EXPRESS DIRECT MEMORY ACCESS - DMA

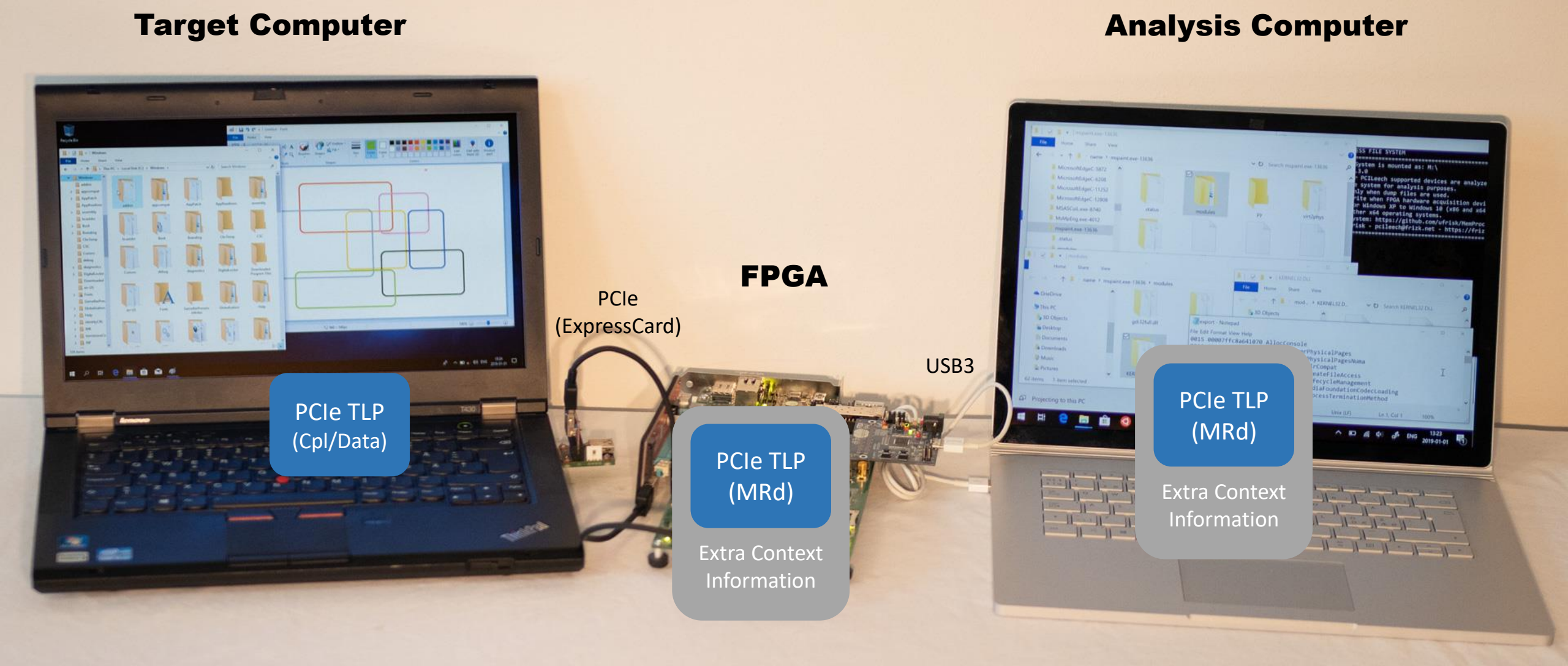
github.com/ufrisk/PCILeech

PCI Express Form Factors



Everything here is PCI Express in different form factors and variations.

Attack & Analysis with DMA HW device



PCILeech

DEMO: WINDOWS

github.com/ufrisk/PCILeech

Windows DEMO:

Execute Code and
Spawn Shell

MOUNT:
target file system

Target System: Windows 10 1903

Windows Security:

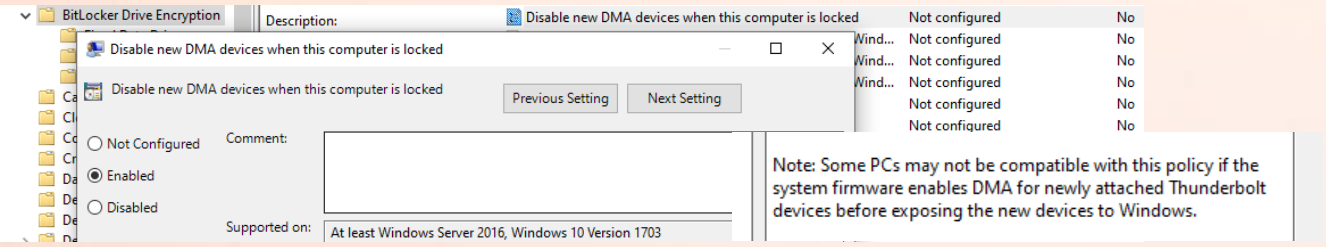
Win7 = vulnerable always!

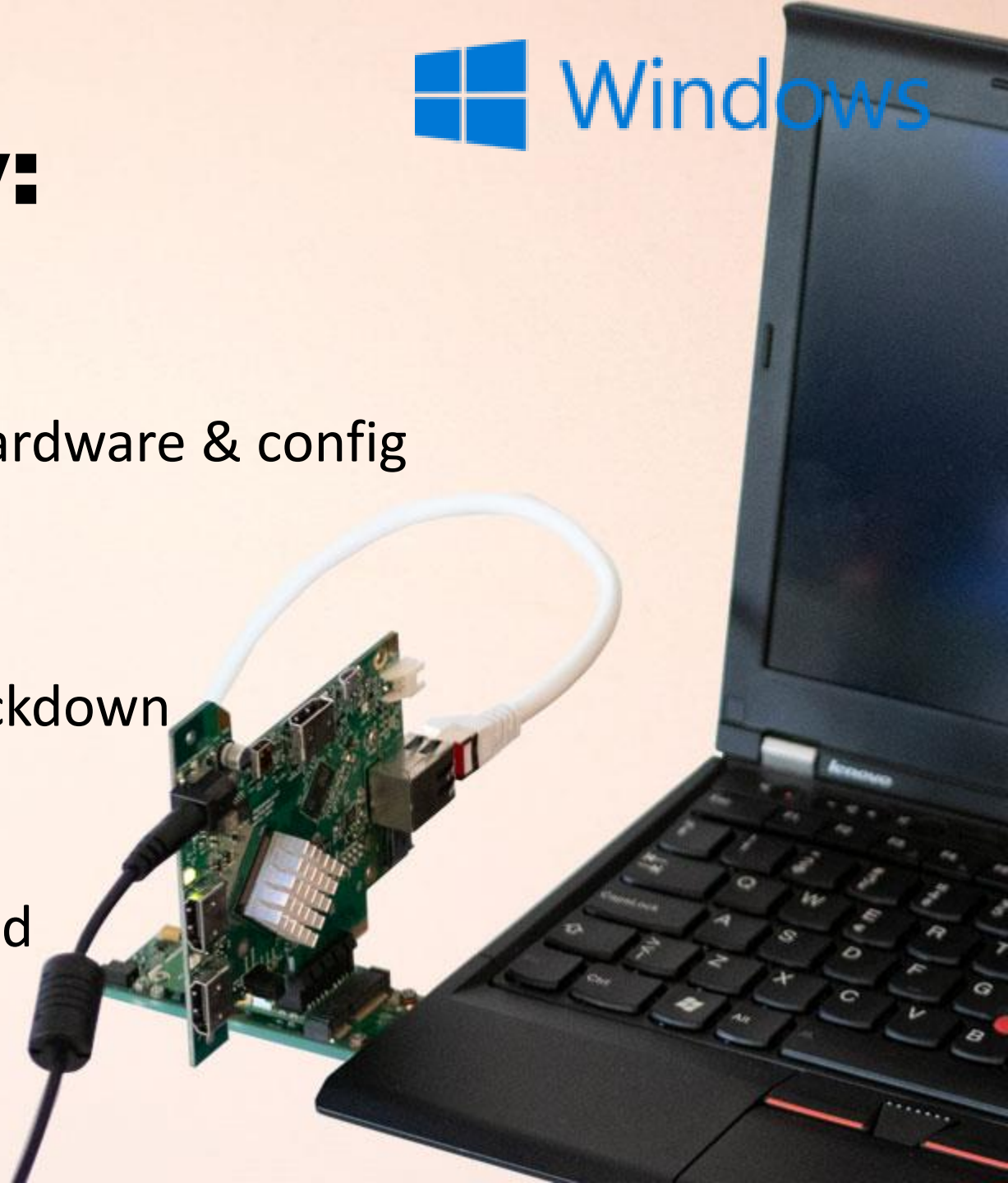
Win10 = (in)secure depending on hardware & config

Hardware without DMA ports

Bitlocker PIN, BIOS password, port lockdown

IOMMU / VT-d,
Virtualization based security,
Disable new DMA devices when locked





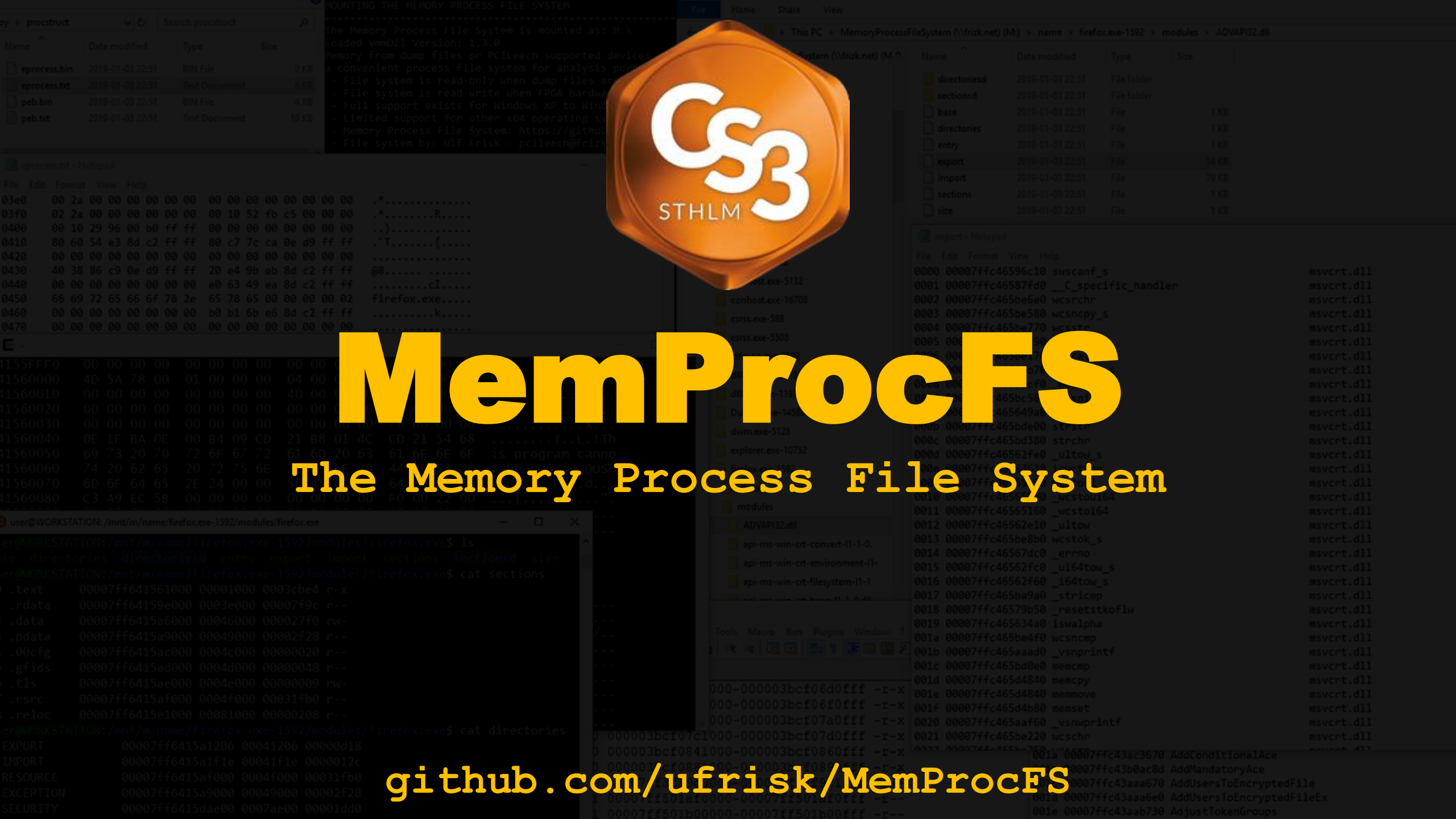
PCILeech: Future

More supported devices

Access via WiFi/4G

Windows User-Mode Implants

Research: macOS/Windows/Linux/ChromeOS/Android/ARM


CS3
STHLM
MemProcFS
The Memory Process File System
github.com/ufrisk/MemProcFS

MemProcFS - The Memory Process File System

Memory Analysis tool with **Windows** focus

In-Memory objects as Files and Folders

Multi-threading + native C core + intelligent parsing → **FAST!**

Wide range of memory acquisition methods: **hardware** and **software**

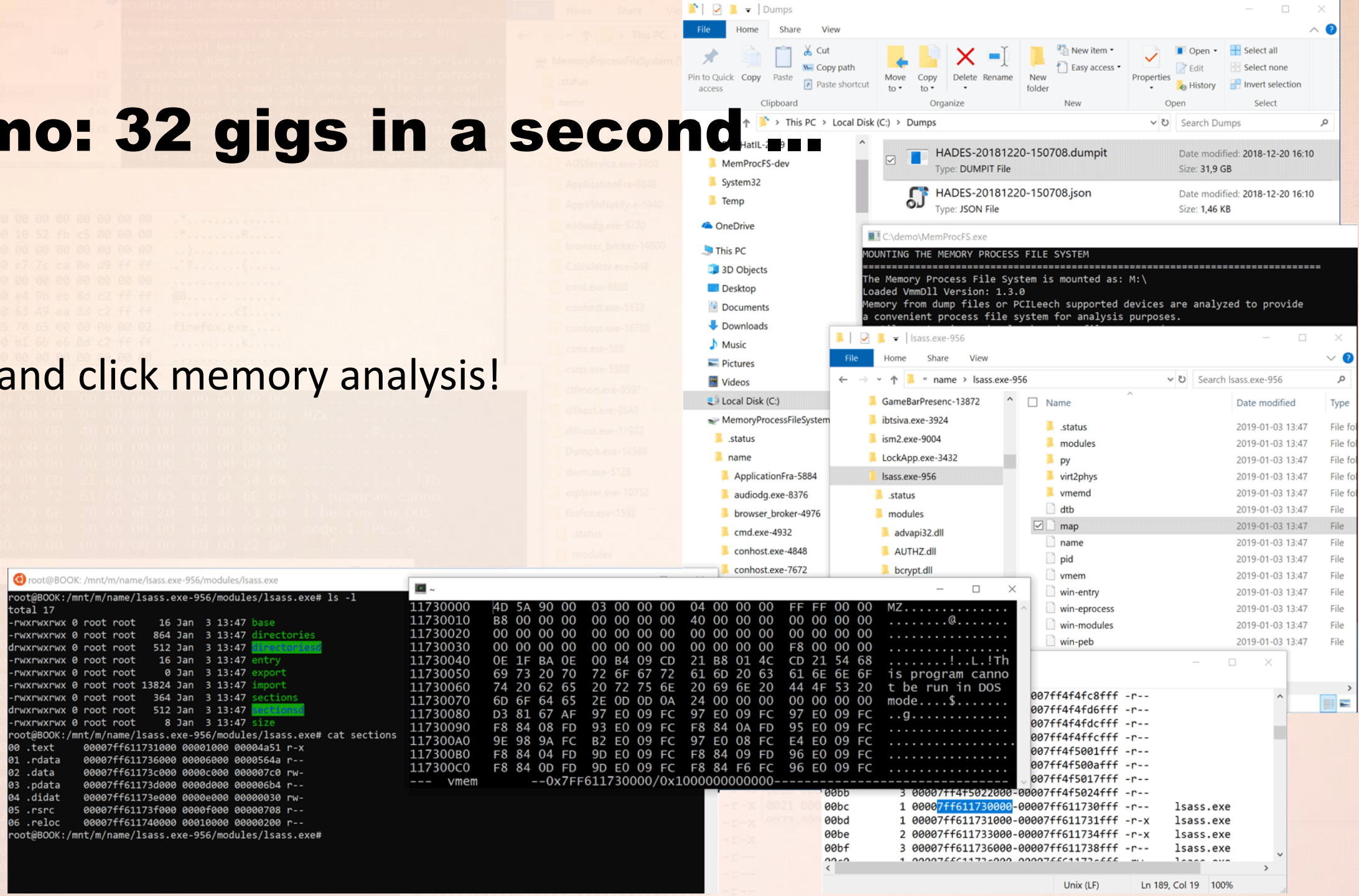
MemProcFS

DEMO: MOUNT 32GB IN A SECOND!

github.com/ufrisk/MemProcFS

Demo: 32 gigs in a second...

Point and click memory analysis!



MemProcFS - Design Goals

Ease of use – but yet powerful

Modular design and plugin functionality

APIs – C and Python

Performance

LeechCore

Memory Acqusition Library

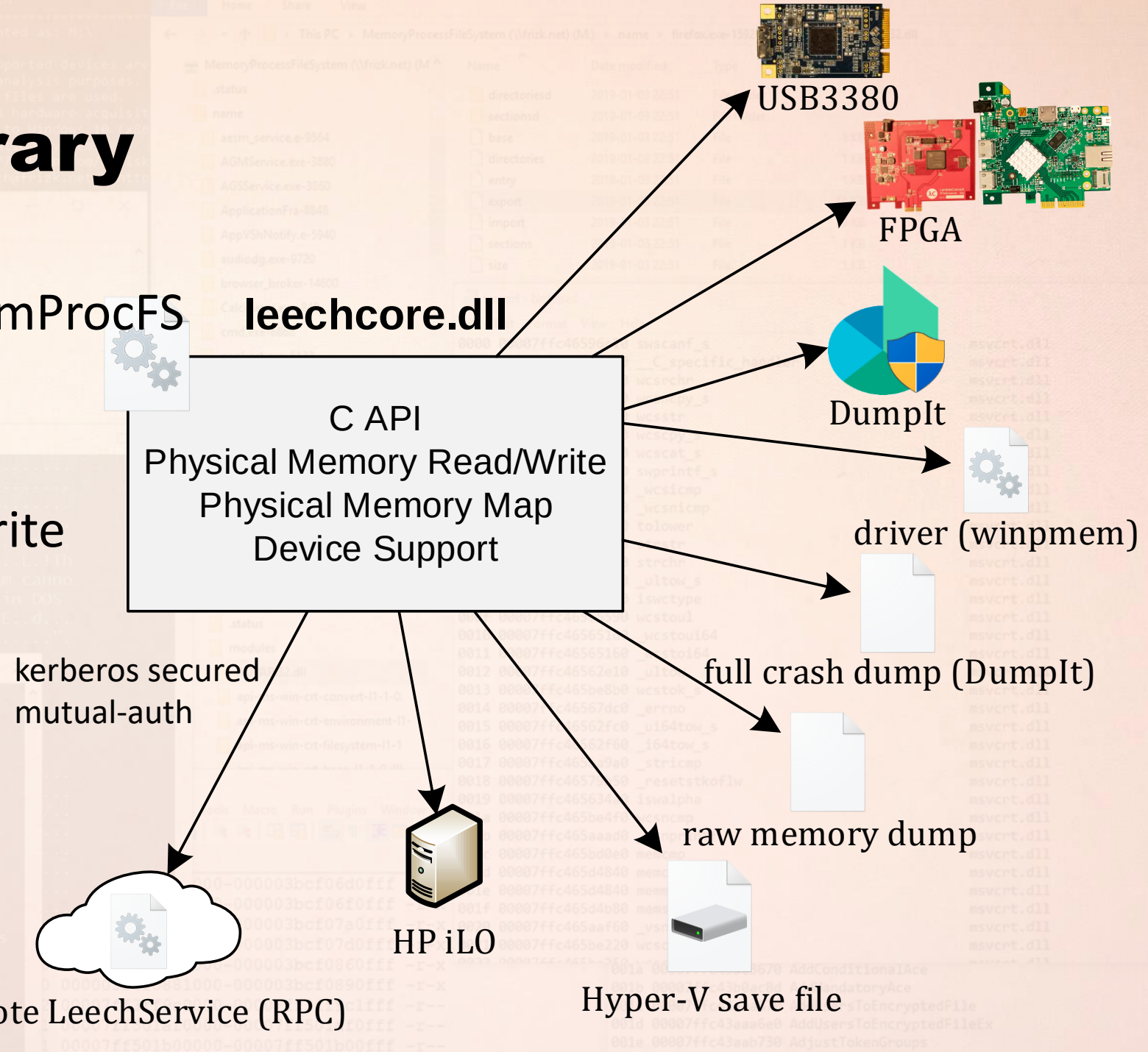
github.com/ufrisk/LeechCore

LeechCore Library

Used by PCILeech and MemProcFS

Focus:
Physical Memory Read/Write

Separates memory acquisition from analysis



MemProcFS

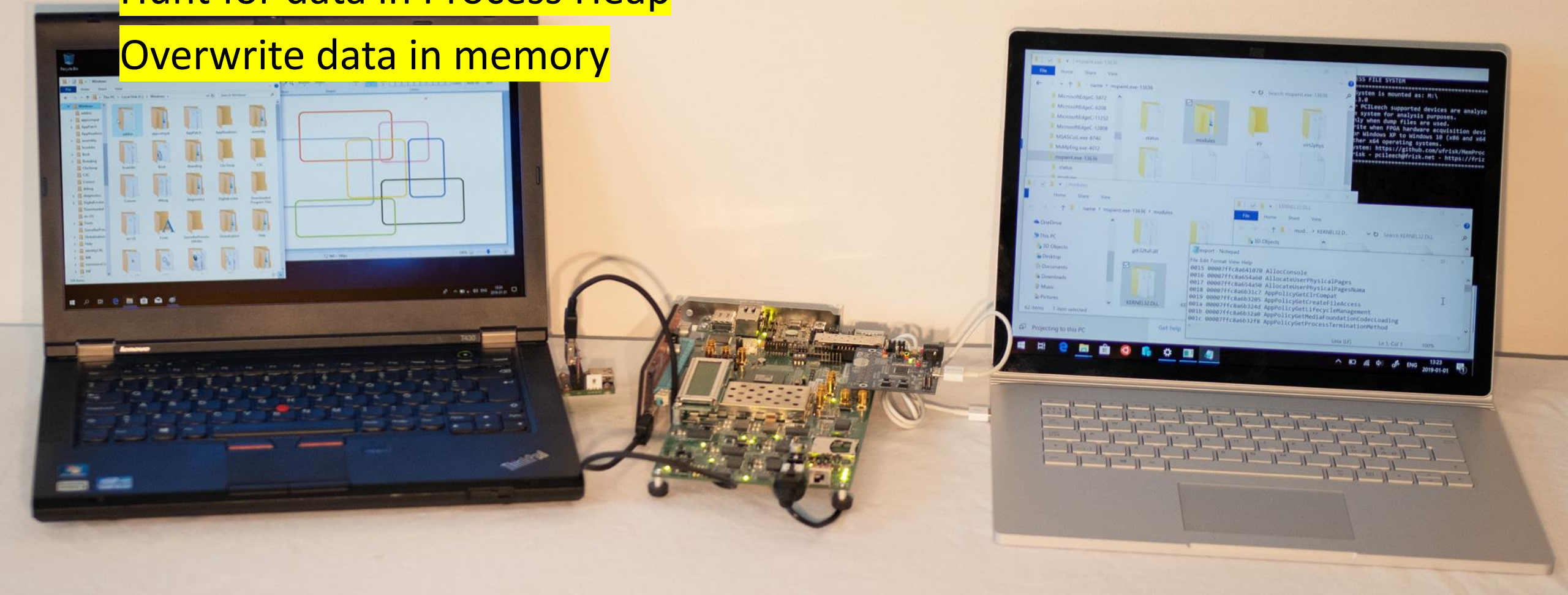
DEMO: LIVE MEMORY ANALYSIS AND ALTERATION

github.com/ufrisk/MemProcFS

Demo: Analysis and write to Live Memory

Hunt for data in Process Heap

Overwrite data in memory



MemProcFS

INCIDENT RESPONSE AND PYTHON API

github.com/ufrisk/MemProcFS

Incident Response with LeechAgent

Suspicious process → Computer Quarantined to VLAN

Limited bandwidth high latency network

Full memory dump == slow

Solution: Retrieve only the memory needed →
Analyze with The Memory Process File System

Or even better … run the **analysis on** the
remote computer by submitting a Python script!

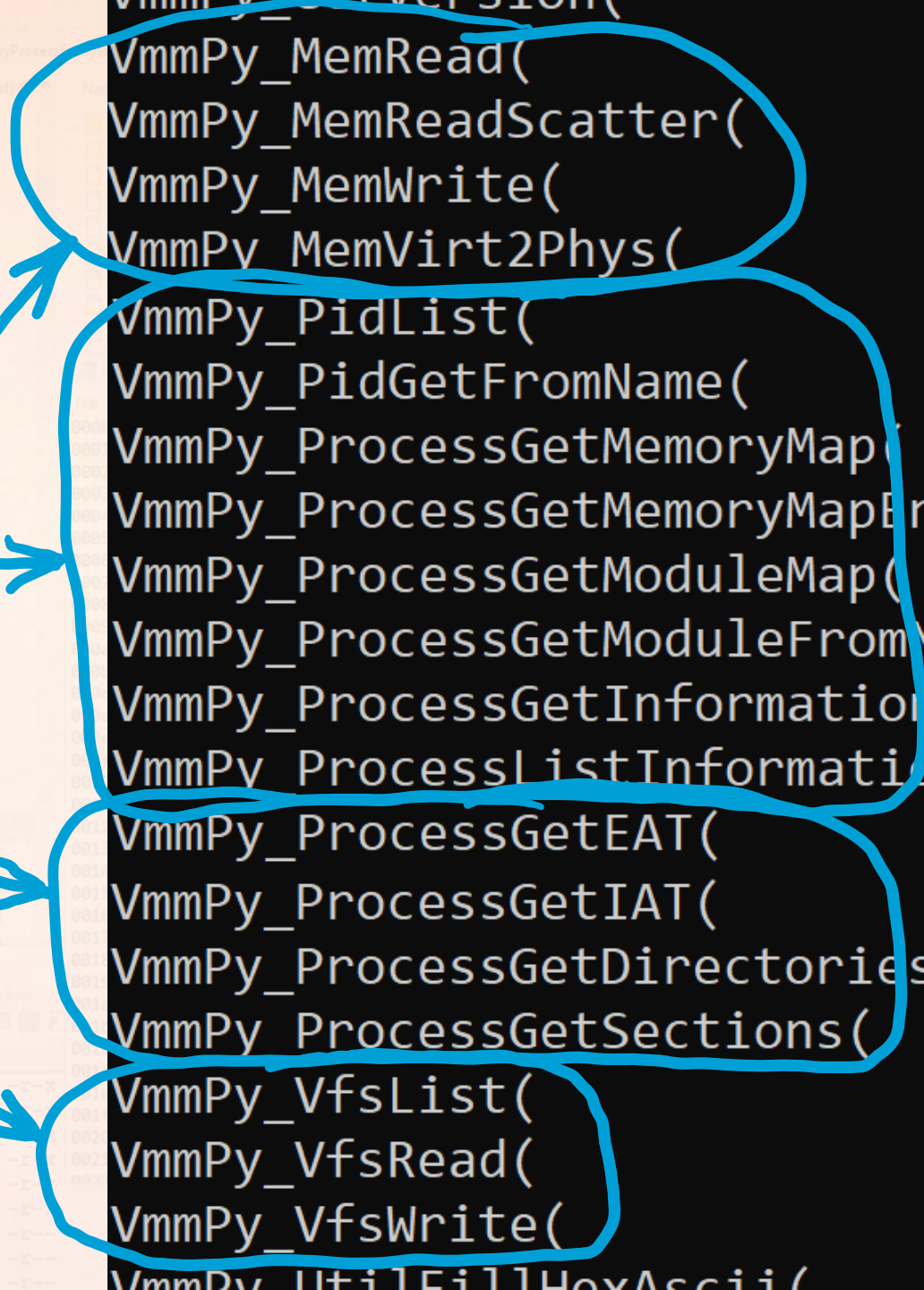
Python API

Read / Write Physical and Virtual Memory

Process information

Modules information

List / Read / Write MemProcFS “files”



MemProcFS

DEMO: REMOTE MALWARE MEMORY ANALYSIS

github.com/ufrisk/MemProcFS

Demo: Remote Malware Memory Analysis

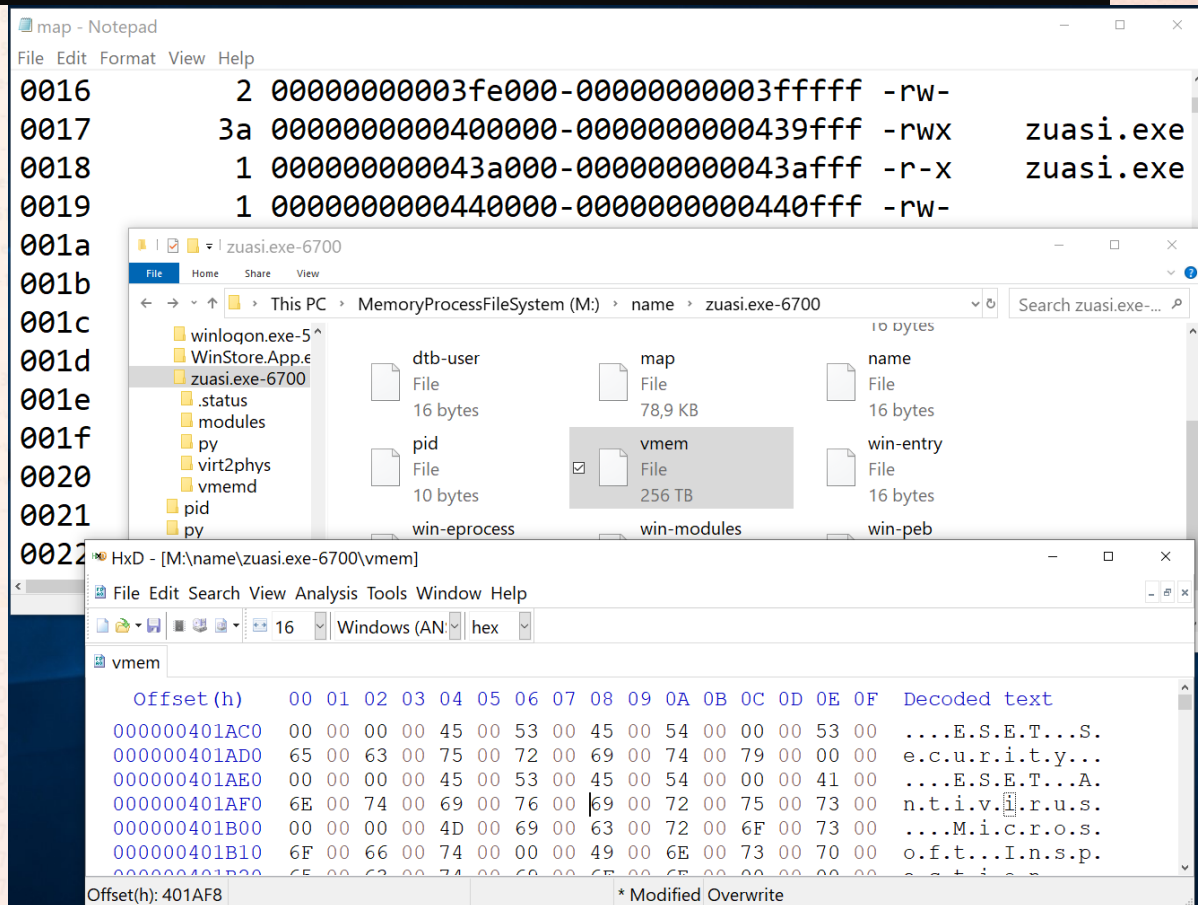
Command Prompt

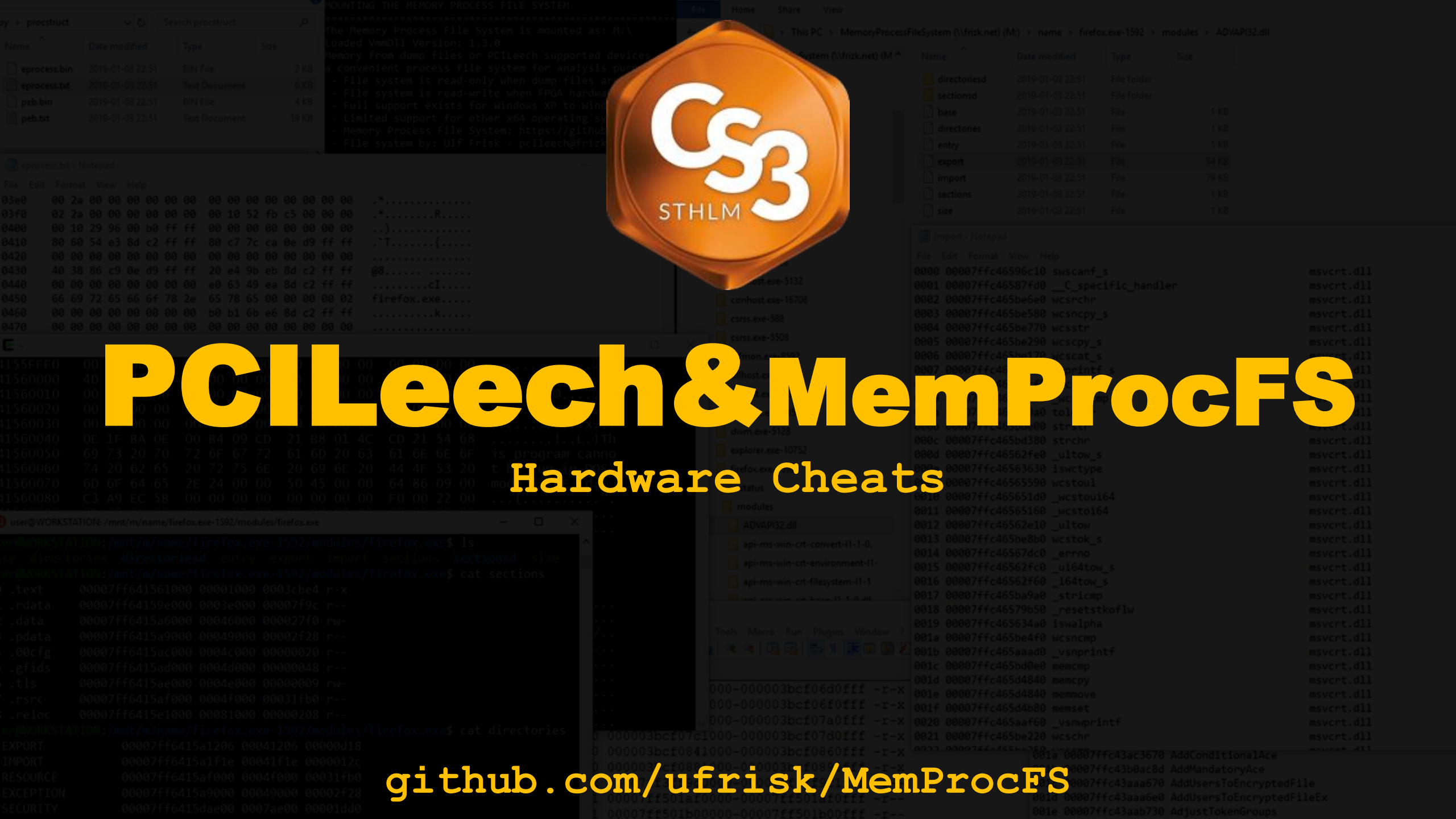
```
Q:\>MemProcFS.exe -device dumpit -remote rpc://kerberos-spn-remote-user:10.9.15.104
```

Analyze **live malware memory**

From **remote** infected system

By clicking on files and using API





PCILeech&MemProcFS

Hardware Cheats

github.com/ufrisk/MemProcFS

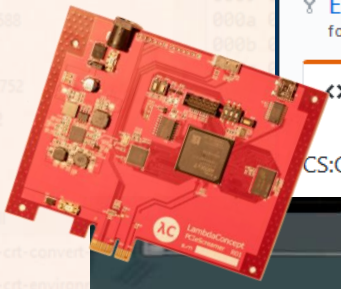
Hardware Cheats with PCILeech/MemProcFS

The **unexpected use case** – cheating in games!

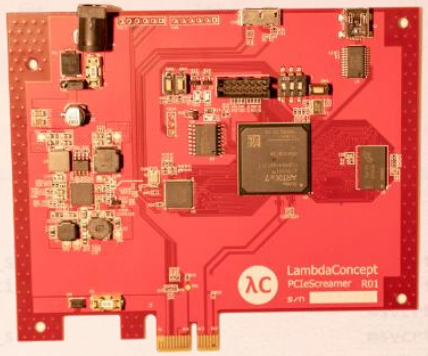
Anti-Cheats – detects software based cheats

HW Cheat – ”only” a PCIe device ...

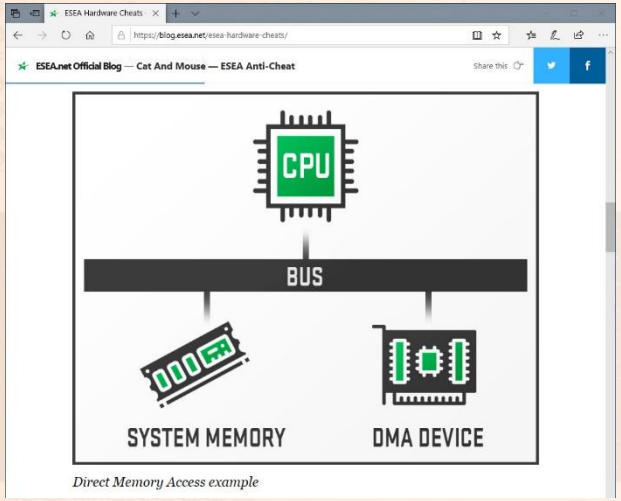
Read-Only ”radar / map decloak”
or Read-Write (more easily detected)



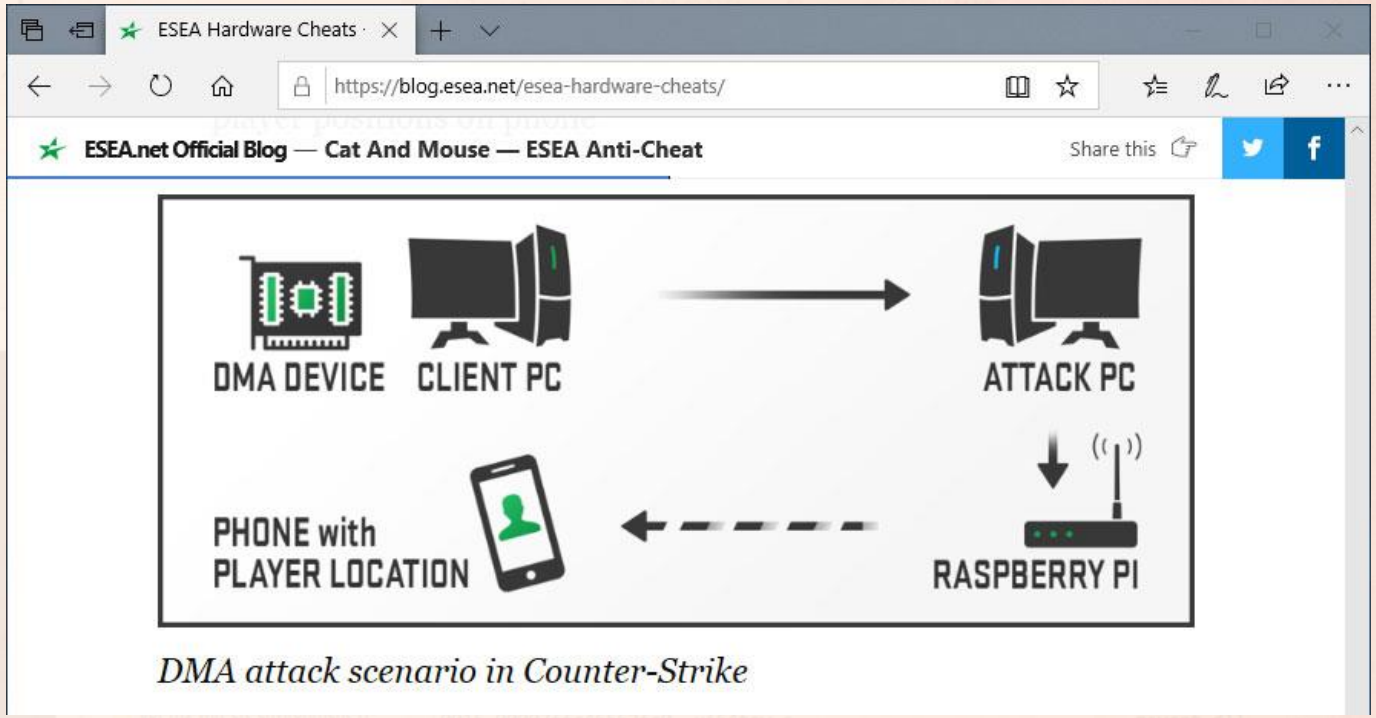
Hardware Cheats



Hardware Cheats



Direct Memory Access example



DMA attack scenario in Counter-Strike

"prices for these cheats have been seen in the **$1,500** to **$5,000** range"

" … **ban wave** of both cheat customers and developers …"

" … can detect hardware-based cheats even when disguising the hardware cheat as a legitimate device. "

MemProcFS Summary

Fast – multi-threaded native C-core

Easy-to-Use – Point and click memory forensics in file system abstraction

Acquisition – Dump Files, Live Memory: driver, hw or remote agent!

Plugin support – roll your own plugins in C/C++/Python!

API – C/C++/Python

github.com/ufrisk/MemProcFS

Thank You!

```
Current Action: Dumping Memory
Access Mode:    KMD (kernel module assisted DMA)
Progress:       8678 / 8678 (100%)
Speed:          173 MB/s
Address:        0x000000021E600000
Pages read:     2050967 / 2221568 (92%)
Pages failed:   170601 (7%)
Memory Dump: Successful.
```

UlfFrisk

github.com/ufrisk