Leveraging Intel DCI for Memory Forensics

By:
Tobias Latzo (Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)), Matti Schulze (FAU), and Felix Freiling (FAU)

From the proceedings of
The Digital Forensic Research Conference
DFRWS USA 2021
July 12-15, 2021

Leveraging Intel DCI for Memory Forensics

Tobias Latzo, Matti Schulze, and Felix Freiling

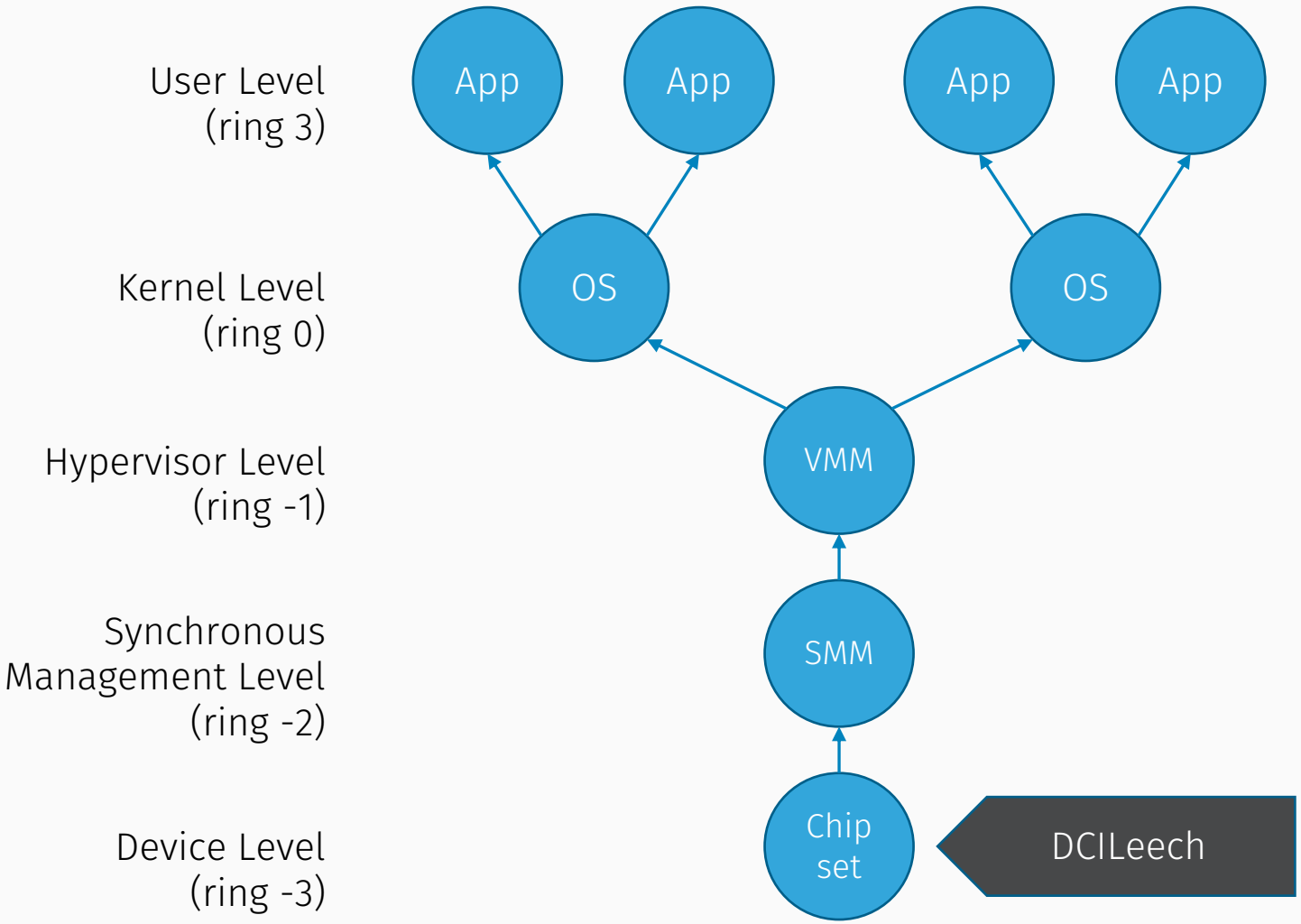
July 12, 2021

Security Research Group
Department of Computer Science
Friedrich-Alexander University Erlangen-Nürnberg (FAU)

Increasing importance of memory forensics

- Running processes
- Encryption keys
- Decrypted data
- Network attached storage
- Fileless malware

Memory access hierarchy[1]



[1] T. Latzo, R. Palutke, and F. Freiling. *A universal taxonomy and survey of forensic memory acquisition techniques*. Digital Investigation, 28(Supplement):56-69, 2019.

Criteria of memory acquisition techniques

Definition by Vömel and Freiling[1]:

- Correctness
  - Actual values when snapshot was taken
- Atomicity
  - No inconsistencies due to interleaving memory acquisition
  - Inconsistencies are frequent and have negative impact on the analysis[2]
- Integrity
  - Content of memory is not changed after investigator decides to take a snapshot

[1] S. Vömel, F. Freiling: Correctness, atomicity, and integrity: Defining criteria for forensically-sound memory acquisition. Digital Investigation. 2012
[2] F. Pagani, Fabio, O. Fedorov, and D. Balzarotti. *Introducing the temporal dimension to memory forensics.* ACM TOPS 22.2, 2019

Intel Direct Connect Interface (DCI)

- Low-cost closed chassis JTAG debugging via an USB 3 A-to-A cable
- Nearly unlimited access to the hardware
- Can often be enabled using hidden firmware flags
- On target side no software required

DCI is very powerful and can even halt the CPU.
Let’s use it for atomic memory dumps.

Previous work

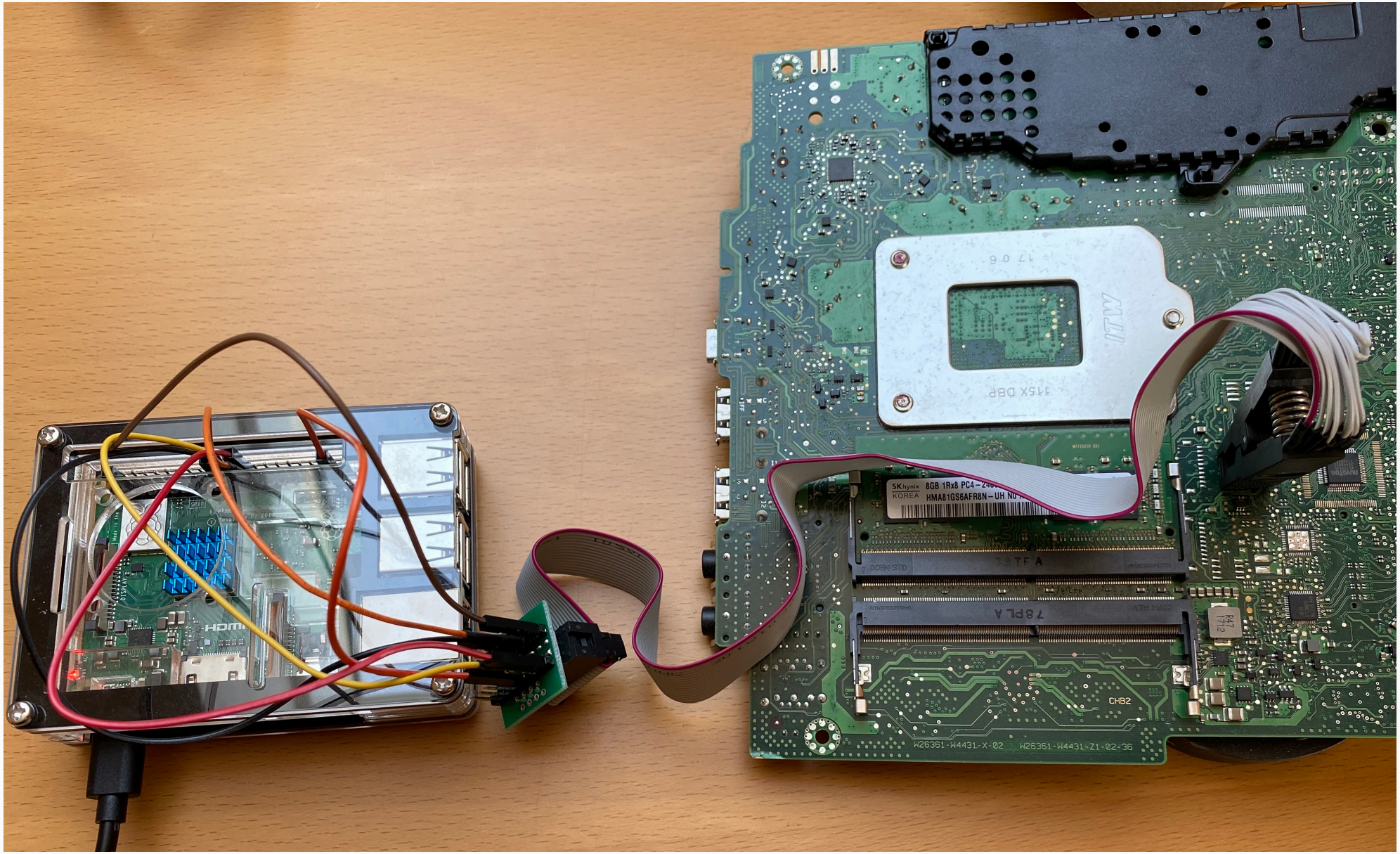
- Most work by Goryachy and Ermolov
  - Tapping into the Core (33c3)
  - Inside Intel Mangement Engine (34c3)
  - Intel DCI Secrets (HITBSecConf 2017)
- Firmware debugging by Jauregui
  - Intro to Closed Chassis Debugging (OSFC 2019)

Activation

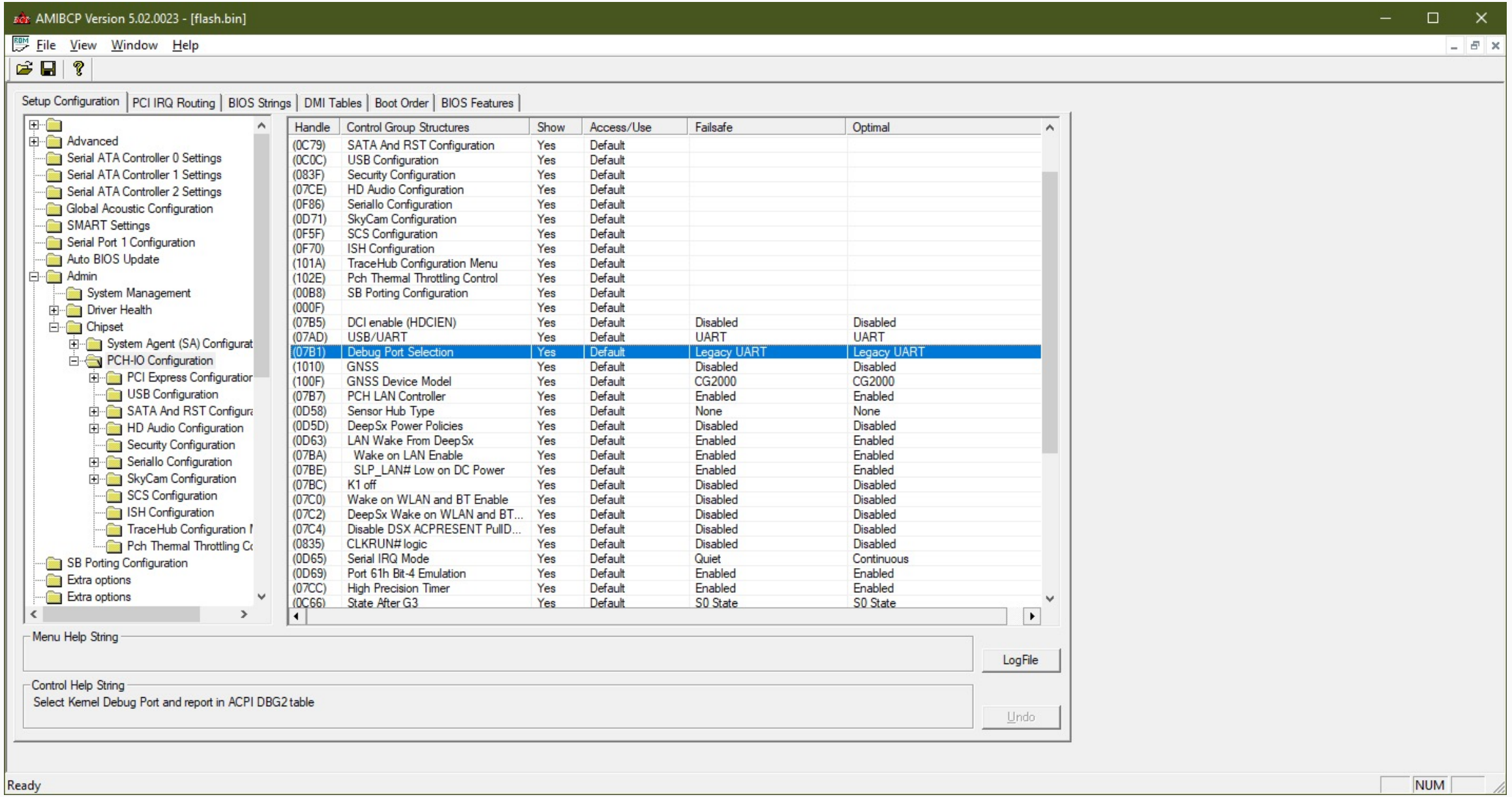
Flags to enable Intel DCI

Flag	Value	Description
Debug Interface	1	Enables silicon debug features
Debug Interface Lock	0	Allows changes of the MSR
Direct Connect Interface	1	Enables DCI
DCI Enable (HDCIEN)	1	Indicates DCI is enabled

Read the EEPROM via SPI



Modifying the firmware flags



DCILeech Implementation

PCILeech

- DMA Attack Framework by Ulf Frisk[1]
- Features:
  - Read and write memory
  - Inject kernel code on the target
  - Push and pull files
  - Shorting Windows login screen
  - Etc.

```
enum rawtcp_cmd {
        STATUS,
        MEM_READ,
        MEM_WRITE
}
```

```
struct rawtcp_msg {
        enum_rawtcp_cmd cmd,
        uint64_t addr,
        uint64_t size
}
```

[1] https://github.com/ufrisk/pcileech

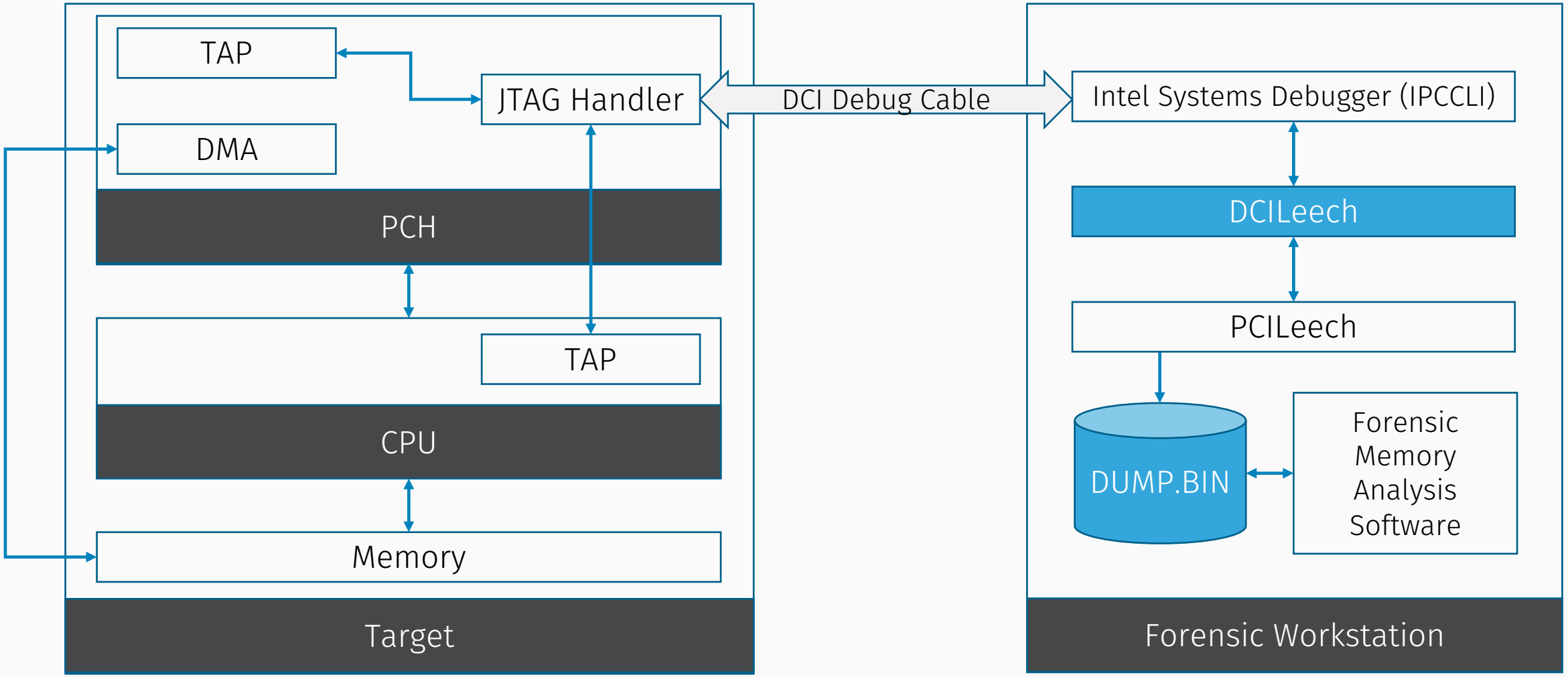
PCILeech extension

Problem: PCILeech expects injected code to be executed

- For many PCILeech features a kernel module is injected
- PCILeech waits for a physical address that is written by the injected code
- However: CPU is halted
- Solution: Extend commands by **GO** and **HALT**

```
enum rawtcp_cmd {
        STATUS,
        MEM_READ,
        MEM_WRITE,
        DCI_GO,
        DCI_HALT
}
```

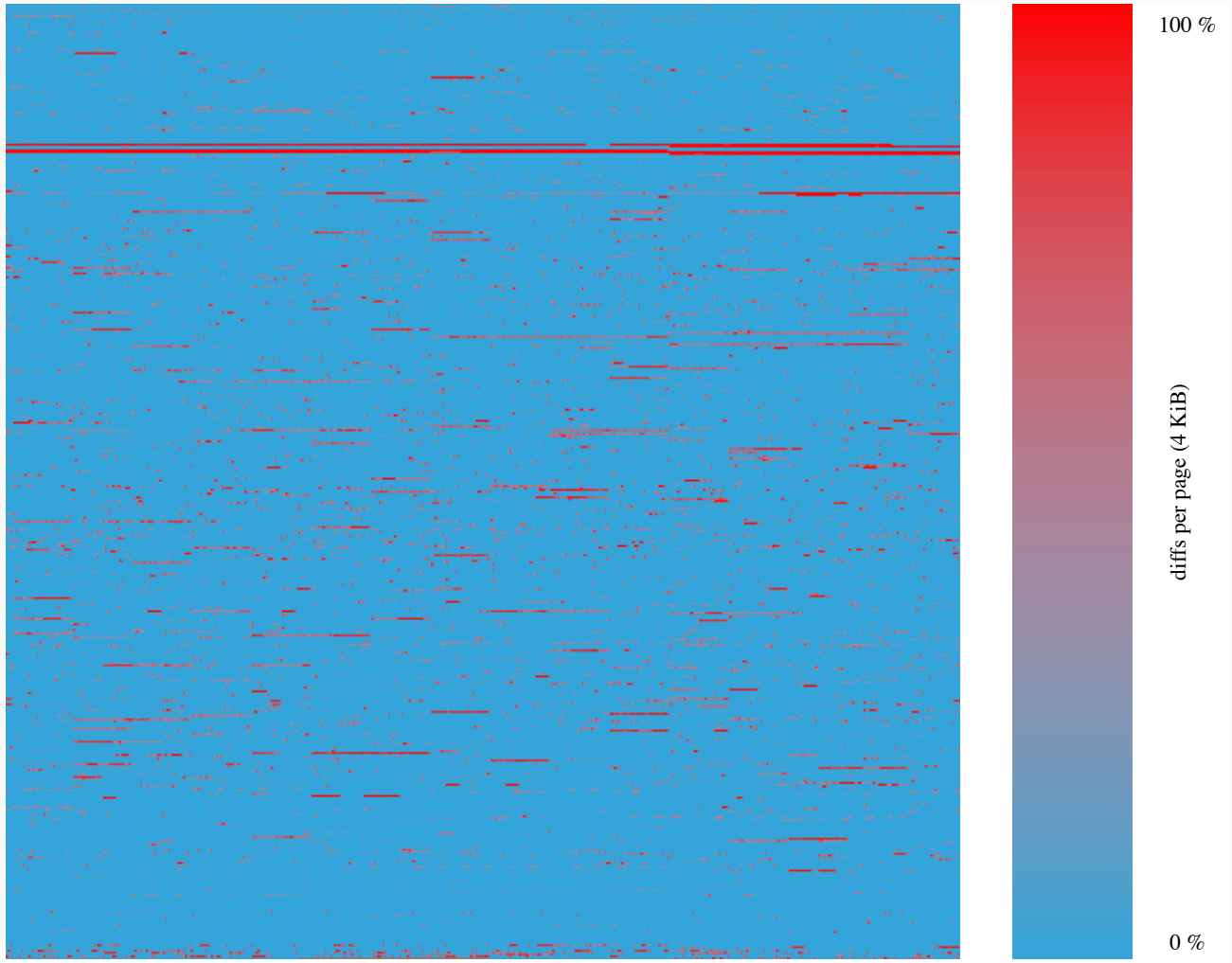
Intel DCI: Architecture



Evaluation

Correctness

Comparison with a LiME dump



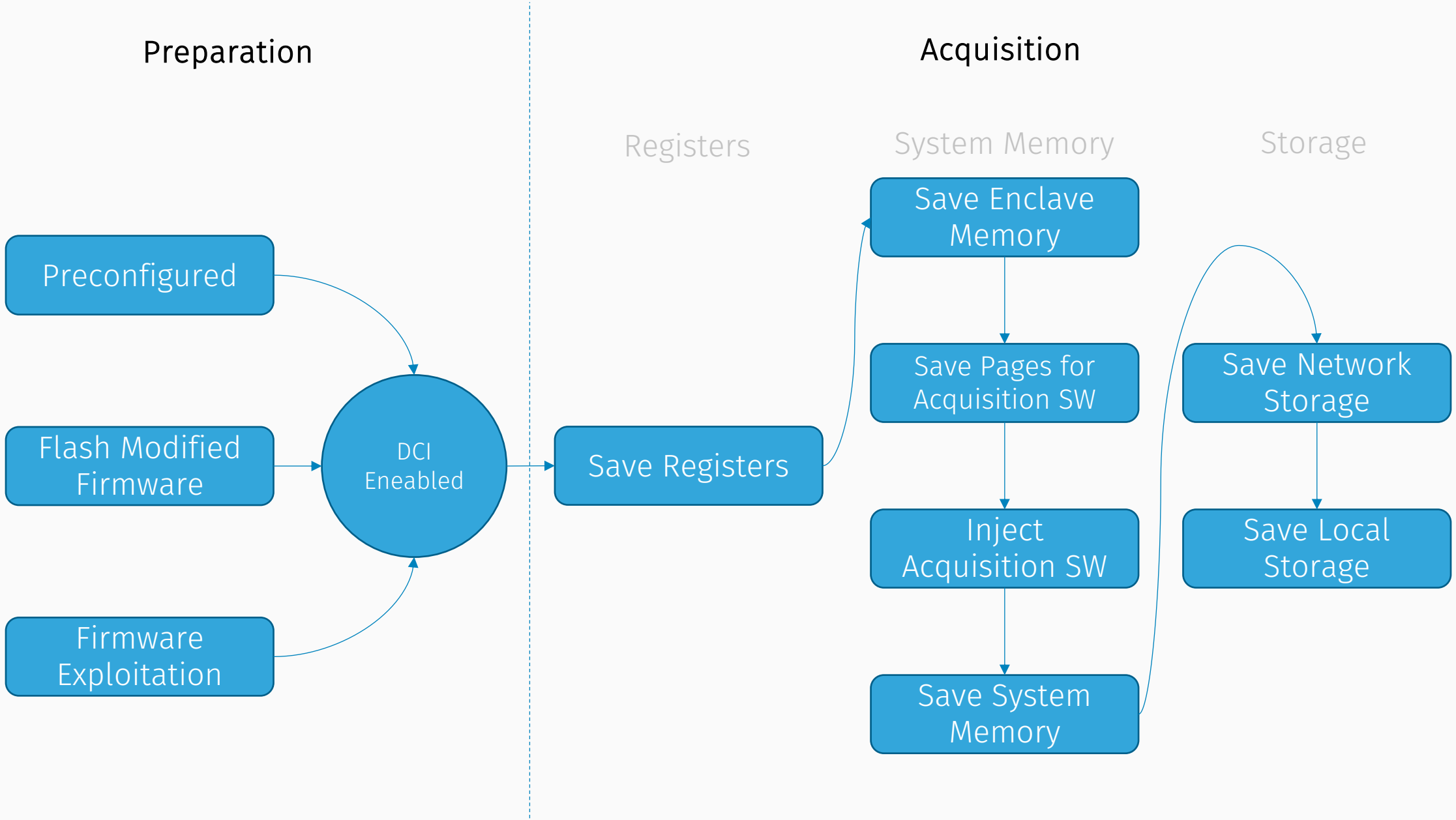
PCILeech payloads

- Memory snapshot: `dump`
- Kernel module injection: `kmdload`
- File retrieval and pushing: `lx64_filepull` and `lx64_filepush`
- Windows 10 unlock: `wx64_unlock`

Intel SGX

- Test program: Load Lena test image into protected enclave memory
- Find address of EPC:
  - `cpuid -l 0x12 -s 0x2`
- Read enclave memory using ITPII: `edbgread`
- Only tested with Debug profile
  - Release profile probably possible with `set_debugoptin`

Forensic Triage with Intel DCI



Conclusion

Conclusion

- Correctness ✓
  - Evaluation showed DCILeech is working properly
- Atomicity: ✓
  - CPU is stopped
- Integrity: ✓
  - No software on target side
- Acquisition of registers possible
- Intel SGX Enclaves can be dumped
  - Debug mode ✓
  - Release mode ?
- Bad performance: 70 KiB/s (SGX 4 KiB/s)
- Limited for on-site investigations → forensic-readiness

Thank you!