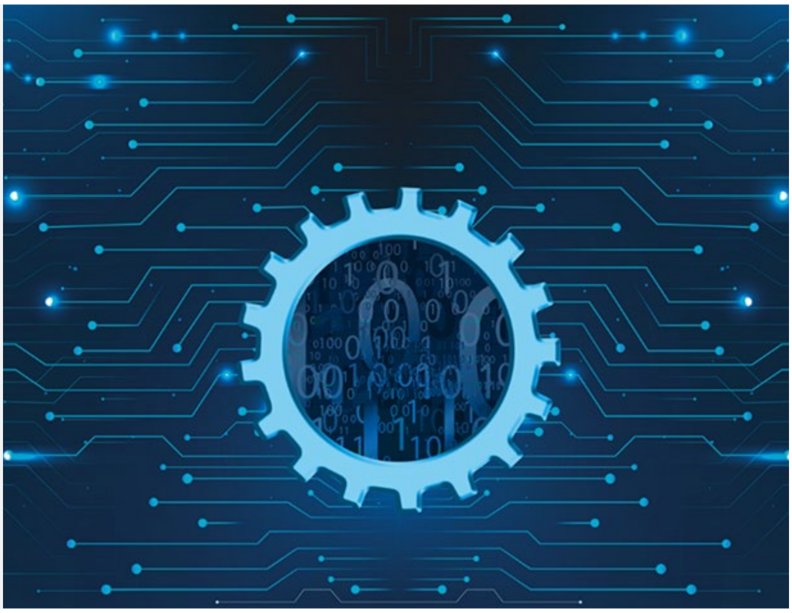
DEPS - https://deps.scch.at
Dependable Production Environments with Software Security

Host: SCCH, www.scch.at

Programme: COMET – Competence Centers for Excellent Technologies

Programme line: COMET-Module



IP PROTECTION USING SELF-ROWHAMMERING CODE

PREVENTING PRODUCT PIRACY THROUGH SOFTWARE-HARDWARE BINDING

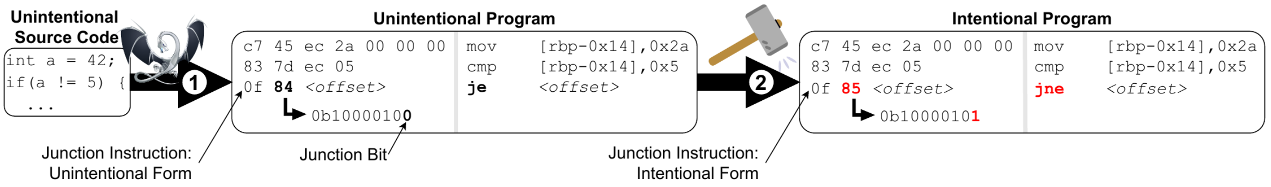
Industrial-scale reverse engineering poses a significant threat, contributing to an annual industry loss estimated at 6.4 billion euros in Germany alone[1]. Typically, hackers that engage in industrial-scale reverse engineering specialize in creating replicas of expensive pieces of machinery, including their software. Presently, their primary efforts involve reverse engineering and replicating the hardware of these machines. Once successful, they can seamlessly run the original software on the cloned hardware without further reverse engineering.

Prior work explored the idea of binding program instances to hardware, ensuring they only function correctly on the designated target machine, and exhibit different behavior on any other (cloned) machine. While this approach creates a level of copy protection, the software still communicates with the machine's Physically Unclonable Function (PUF) through well-defined and easily identifiable interfaces. A potential attacker could exploit this and tamper with the software and eliminate any machine dependence.

In DEPS, we investigate new approaches to prevent the described IP theft. Our work binds hardware and software using PUFs *and* adds stealthiness into the interaction between the software and hardware to make reverse engineering of the protected software a daunting, almost impossible task.

We developed a sophisticated approach leveraging the rowhammer effect to self-modify the code, and simultaneously profit from the stealthy nature of the effect. The rowhammer effect is a hardware disturbance error in DRAM modules that allows unprivileged actors to flip bits in physical memory. Because of implicit electromagnetic couplings in the DRAM chip, one can flip the logical value in a memory cell by accessing a certain pattern of neighboring DRAM cells at a very high frequency, using only standard memory access instructions. Rowhammer's bit flip pattern de-

[1] https://www.vdma.org/documents/34570/51629660/VDMA+Study+Product+Piracy+2022_final.pdf


Unintentional Source Code
Unintentional Program
Intentional Program
1
2
Junction Instruction: Unintentional Form
Junction Bit
Junction Instruction: Intentional Form

```
int a = 42;
if(a != 5) {
  ...
```

```
c7 45 ec 2a 00 00 00   mov   [rbp-0x14],0x2a
83 7d ec 05            cmp   [rbp-0x14],0x5
0f 84 <offset>         je    <offset>
   0b10000100
```

```
c7 45 ec 2a 00 00 00   mov   [rbp-0x14],0x2a
83 7d ec 05            cmp   [rbp-0x14],0x5
0f 85 <offset>         jne   <offset>
   0b10000101
```

pends on minor, unavoidable variations in the manufacturing process which makes it unclonable and practically unique for each DRAM instance. Earlier research already leveraged this feature to use it as a PUF.

Objectives

Our envisioned system should:

- provide efficient and secure protection to a wide class of industrial software.
- bind a protected software instance to a target machine in a way that the program behaves correctly only on the target machine, and differently on any other (cloned) machine.
- reconstruct the desired program at run time based on unclonable and machine-specific behavior without the need for external hard-ware.
- make reverse engineering unfeasible or uneconomical by hiding any information regarding the desired behavior of the protected program from the binary image and during its execution on any machine other than the target machine.

DEPS Scientific Answer

We developed a system that creates programs that self-modify their code section to change their behavior using rowhammer-induced bit flips.

The developer changes the source code to model the unintentional behavior and outlines all required program changes to reconstruct the intended behavior. Our custom toolchain takes in this information, scans the target machine for rowhammer-susceptible cells, and subsequently builds the software with a custom layout to facilitate the required pro-gram changes using rowhammer-susceptible DRAM cells. Subsequently, the protected program performs a row-hammer attack on its own code region to induce bit flips. The key property is that the bit flips required to reconstruct the intentional program will only manifest themselves on the target machine and not on any cloned machinery. This is due to the locations of the rowhammer-susceptible cells, which were used during compilation, being machine-specific and un-clonable.

Furthermore, the use of rowhammer brings several advantageous properties, among which:

- the rowhammer effect is inherently present in all DRAM modules of the last fourteen years and, therefore, widely available on nearly all computing devices.
- the bit flip pattern is unclonable and practically unique.
- the location of the rowhammer-susceptible cells is statically unknown.
- bit flips, and therefore, program modifications, are invisible to the OS and the CPU.

These properties allow for stealthy program modifications that, even on the target machine, cannot be monitored without constant memory scans at regular, short time intervals. Additionally, the binary image contains no traces of the intentional program behavior and does not leak information when executed on any other than the target machine. By employing rowhammer as a PUF, we achieve our aforementioned objectives.

Our evaluation indicates that the current state of our system has an acceptable performance overhead.

Related DEPS Publications

[1] R. Mechelinck, D. Dorfmeister, B. Fischer, S. Volckaert, and S. Brunthaler. "Exploring Rowhammer-Based Cooperative Weird Machines for Stealthy Computation." Under Submission.

Contact: Juliana Küster Filipe Bowles, DEPS Coordinator, SCCH, T +43 50 343 900, juliana.bowles@scch.at

Consortium:

Federal Ministry
Republic of Austria
Climate Action, Environment,
Energy, Mobility,
Innovation and Technology

Federal Ministry
Republic of Austria
Labour and Economy

FFG
Promoting Innovation.

Austrian Research Promotion Agency
Sensengasse 1, A-1090 Vienna
P +43 (0) 5 77 55 - 0
office@ffg.at
www.ffg.at