Randomization can't stop BPF JIT spray

Elena Reshetova, N. Asokan

Aalto University

Collaborative Research Institute for Secure Computing

Berkeley Packet Filter (BPF) JIT spray attack was first introduced in 2012

Upstream Linux kernel fix has held till 2016 despite concerns

We show that the fix is vulnerable to a new modified attack

BPFs now used in Linux not just for packet filtering: various tracepoints and seccomp. It has been called a "universal in-kernel virtual machine"

Original JIT spray attack against BPF JIT (2012)

- Pass payload instructions as constants in different BPF instructions
- Populate address space with many filters
- Randomly guess filter start page and jump to it

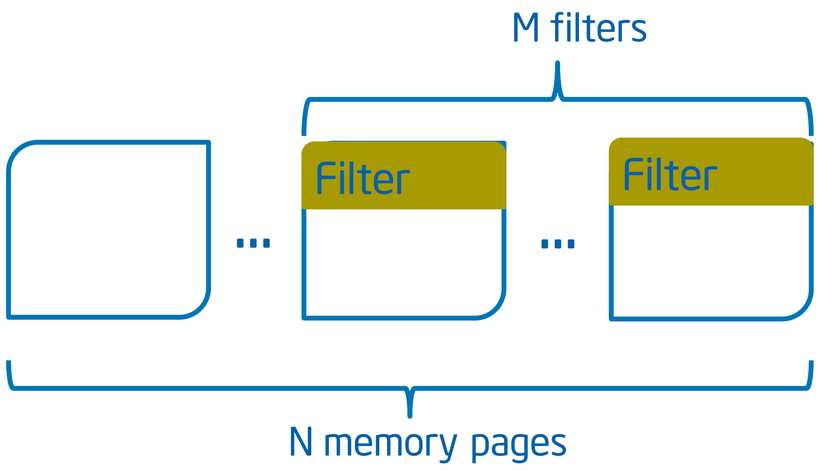

Fig 1: Original BPF JIT spray attack: Attacker needs to guess a page with a filter. Bad guess not punished.

Community reaction

- Grsecurity: blind constants in BPF instructions
- Upstream kernel: randomize BPF start address
- No attack against upstream fix until now

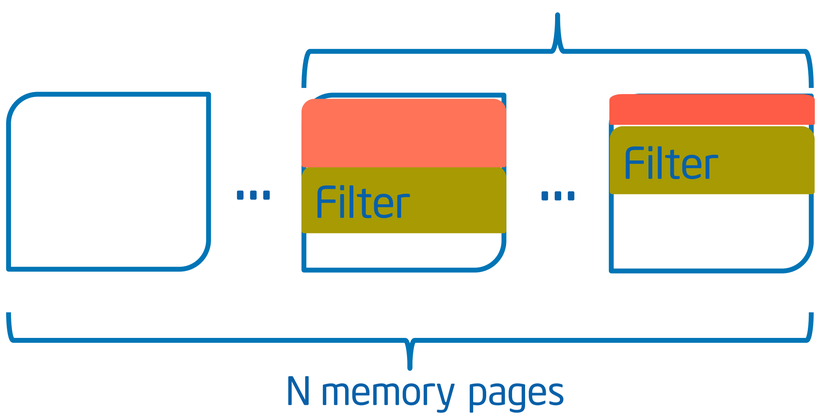

Fig 2: Original fix: Attacker needs to guess a page with a filter and an offset. Bad guess will likely be punished.

Our attack

- Repeat payload enough times for filter to grow beyond one page
- Guess random page but try executing all offsets (1-10) at page start
- Adding enough "NOPs" before payload start minimizes attack failure during execution attempts

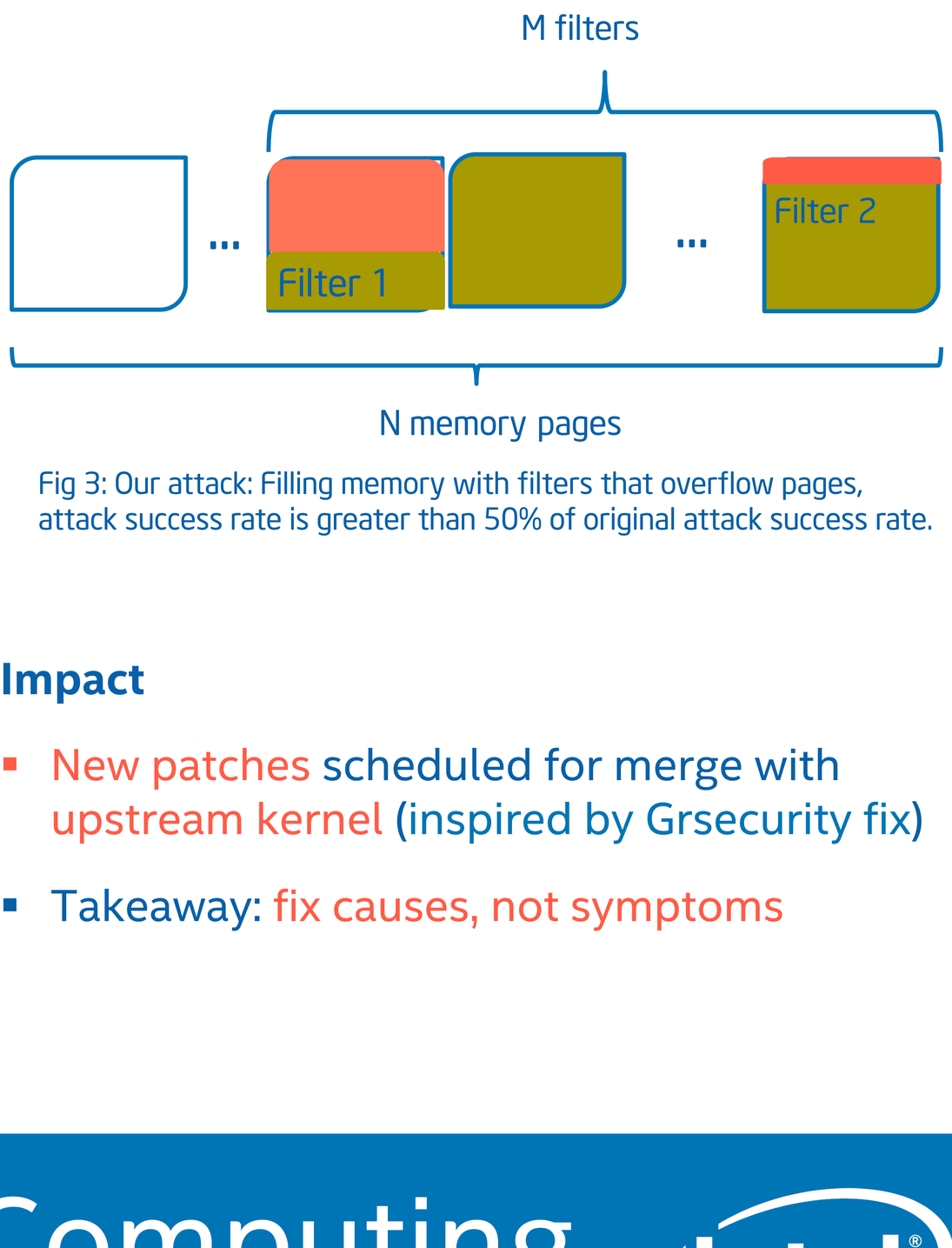

Fig 3: Our attack: Filling memory with filters that overflow pages, attack success rate is greater than 50% of original attack success rate.

Impact

- New patches scheduled for merge with upstream kernel (inspired by Grsecurity fix)
- Takeaway: fix causes, not symptoms

Intel CRI for Secure Computing

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
*Other names and brands may be claimed as the property of others.