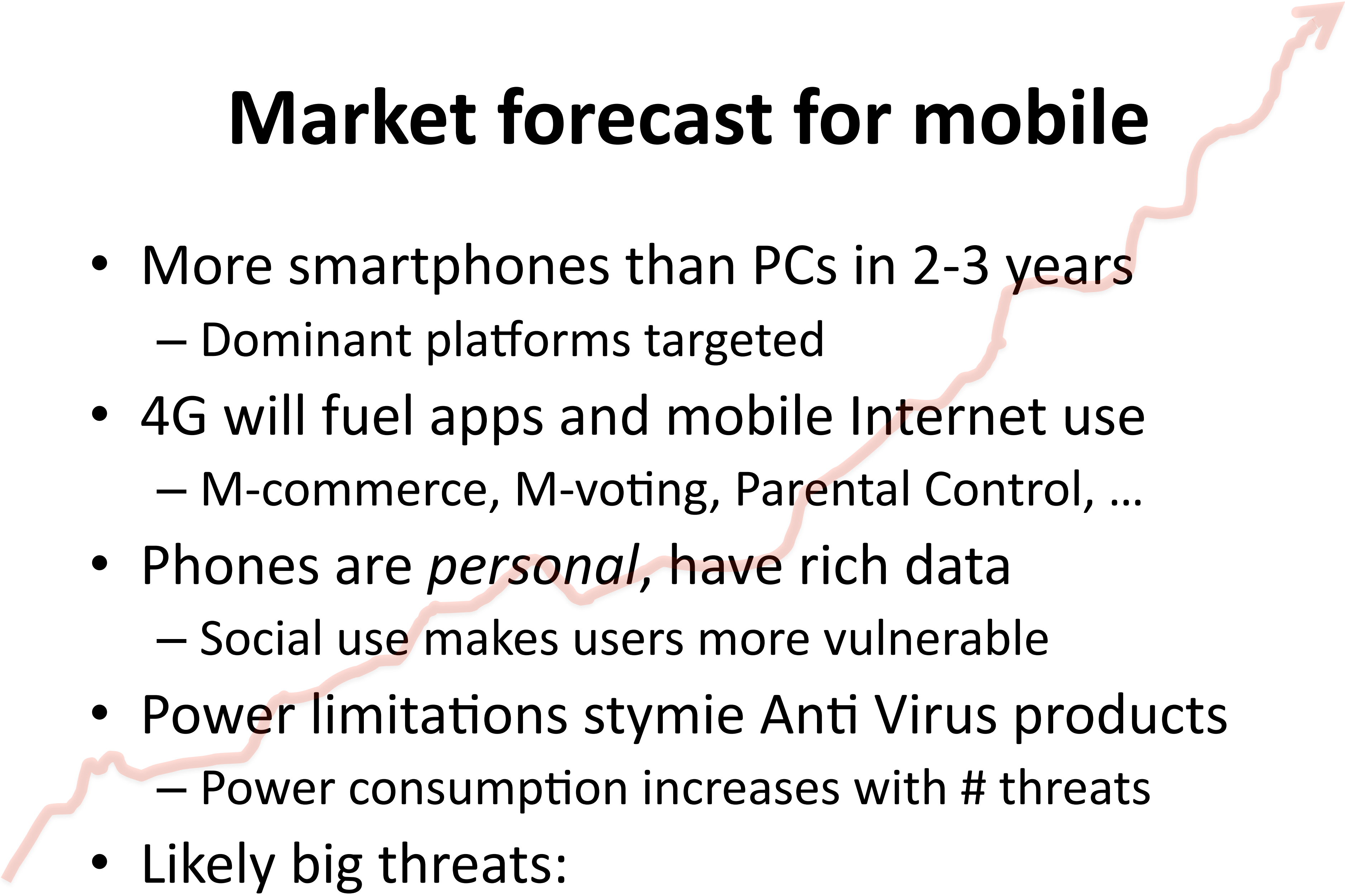
Retroactive Detection of Malware with Applications to Mobile Platforms

Markus Jakobsson Karl-Anders Johansson

FatSkunk

Market forecast for mobile



- More smartphones than PCs in 2-3 years
  - Dominant platforms targeted
- 4G will fuel apps and mobile Internet use
  - M-commerce, M-voting, Parental Control, …
- Phones are *personal*, have rich data
  - Social use makes users more vulnerable
- Power limitations stymie Anti Virus products
  - Power consumption increases with # threats
- Likely big threats:
  - Bluetooth viruses, (piracy) trojans, social malware

Trends: Faster, stealthier, smarter

kits, recompilers, polymorphism

malware often installs AV (limit competition)

produced by organized crime

Contrast: What the consumer wants

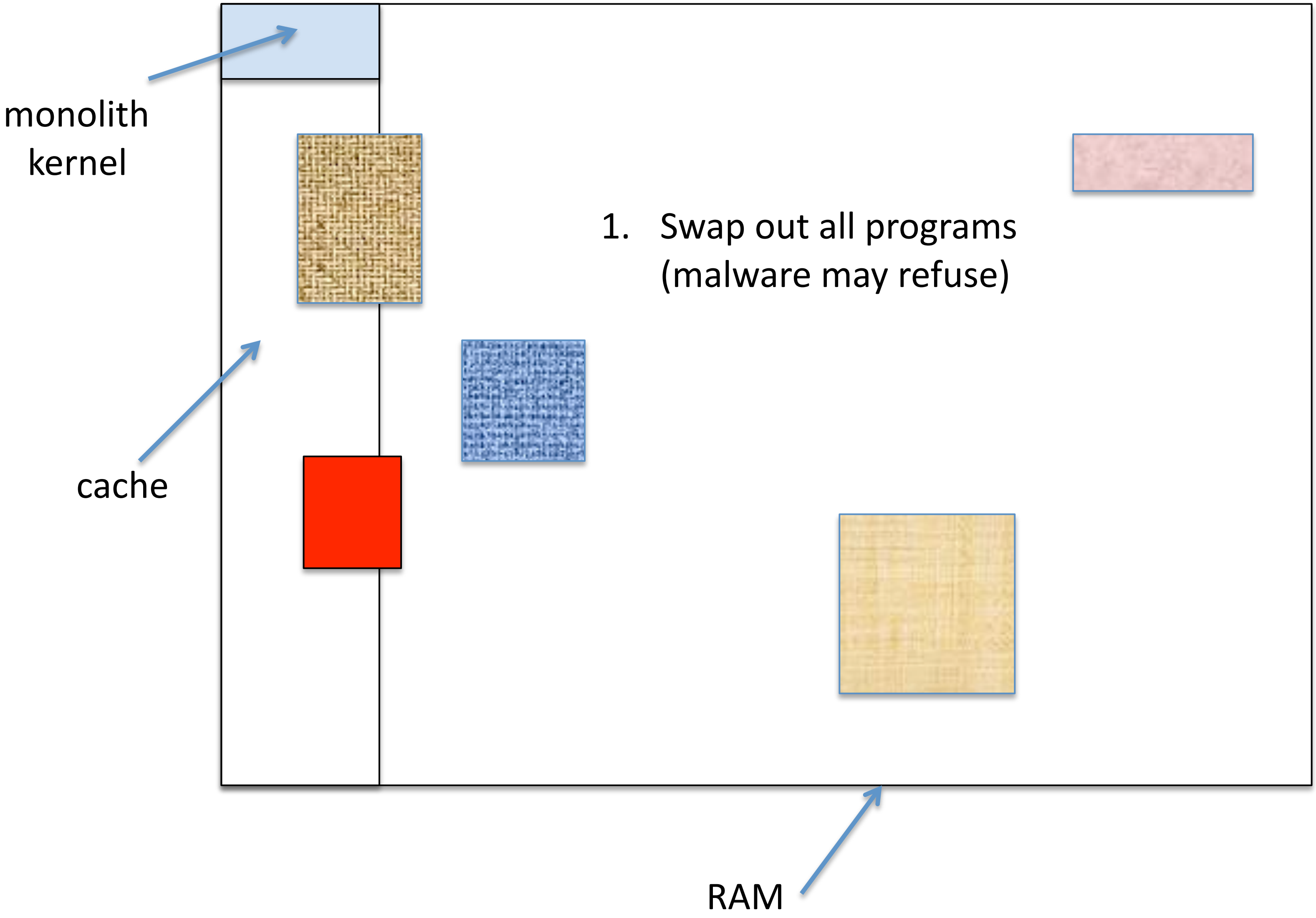
Undo

What makes this challenging

1. Malware masquerades and deceives
2. Malware will not allow itself be erased
3. Malware can catch interrupts
4. Malware can edit system calls/responses
5. *Malware is bad, will not cooperate*

Main principles

- To block detection, malware must be *active*.
- To be active, malware needs to *be in RAM*.
- RAM is *faster* than flash and radio.



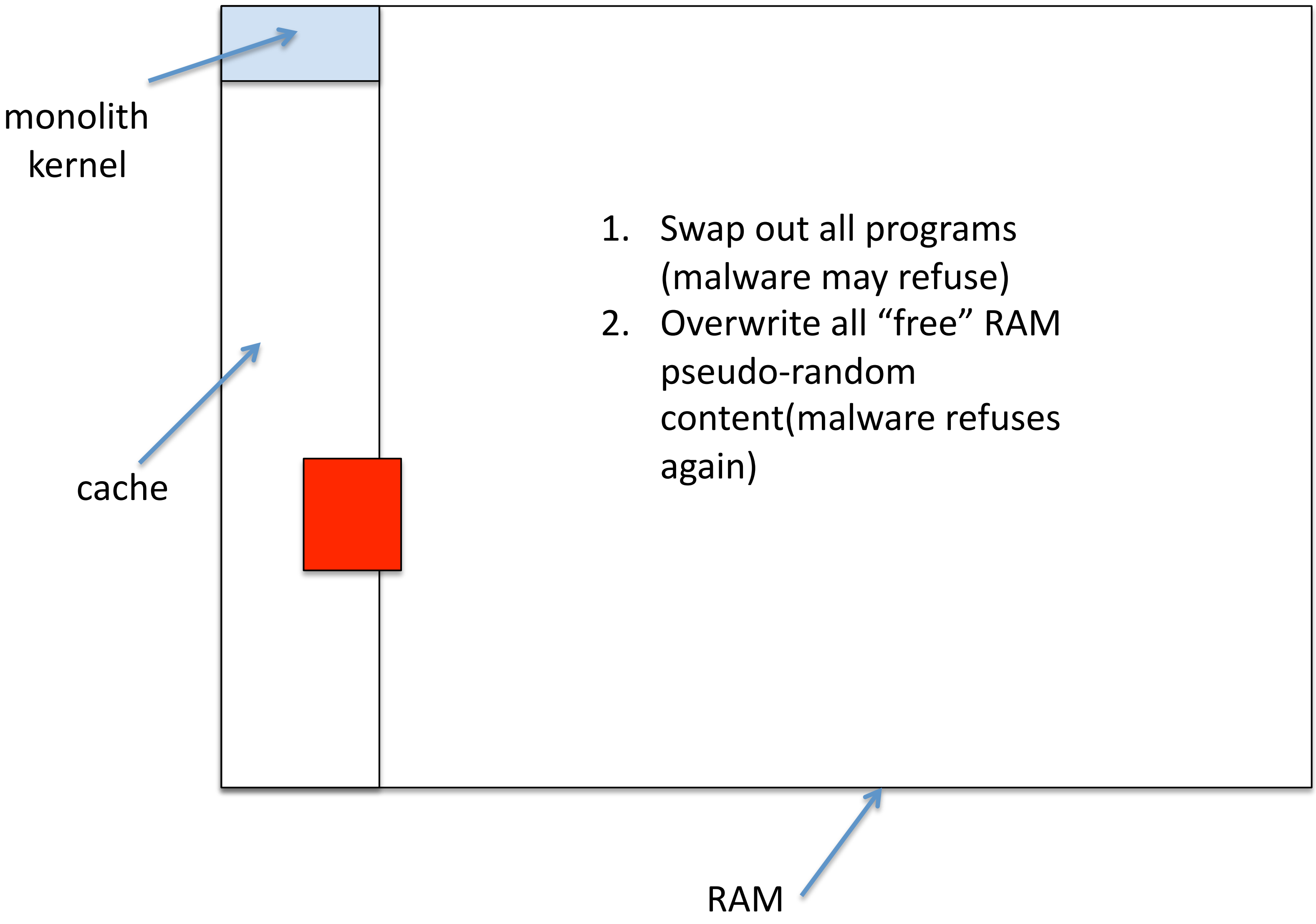
monolith kernel

cache

RAM

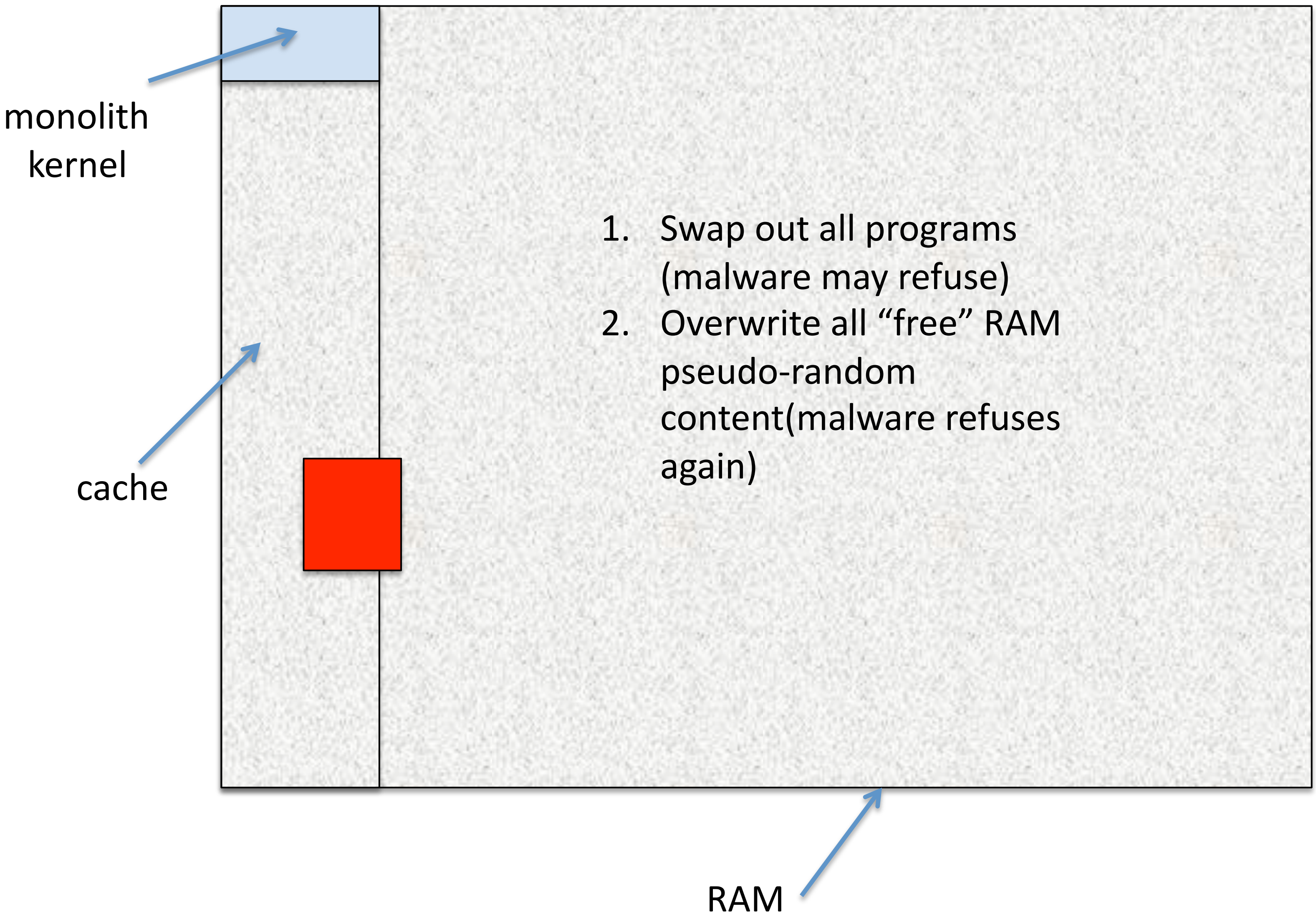
1. Swap out all programs (malware may refuse)

Contact markus@fatskunk.com for more details incl. improvements.



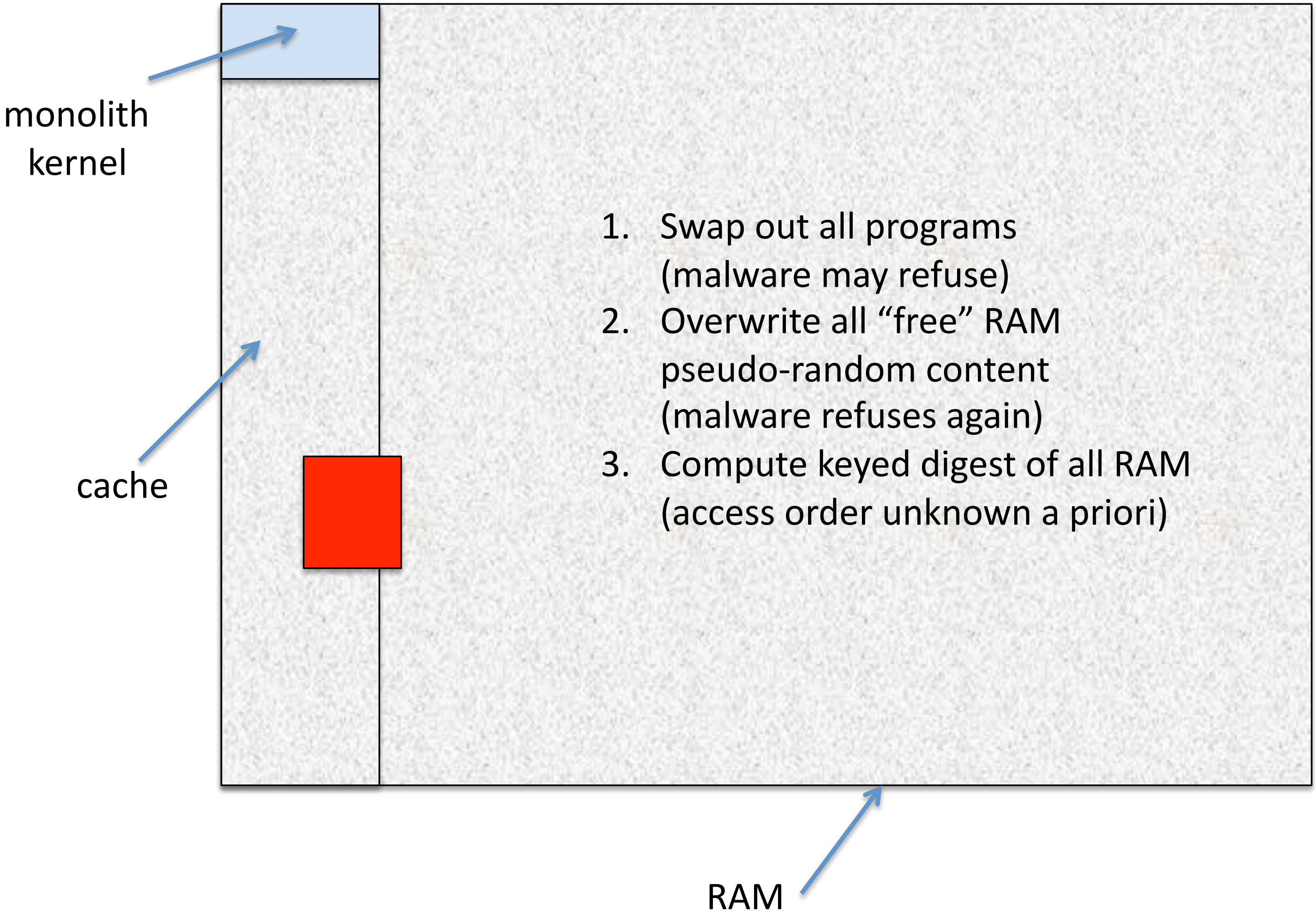
1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content(malware refuses again)

Contact markus@fatskunk.com for more details incl. improvements.



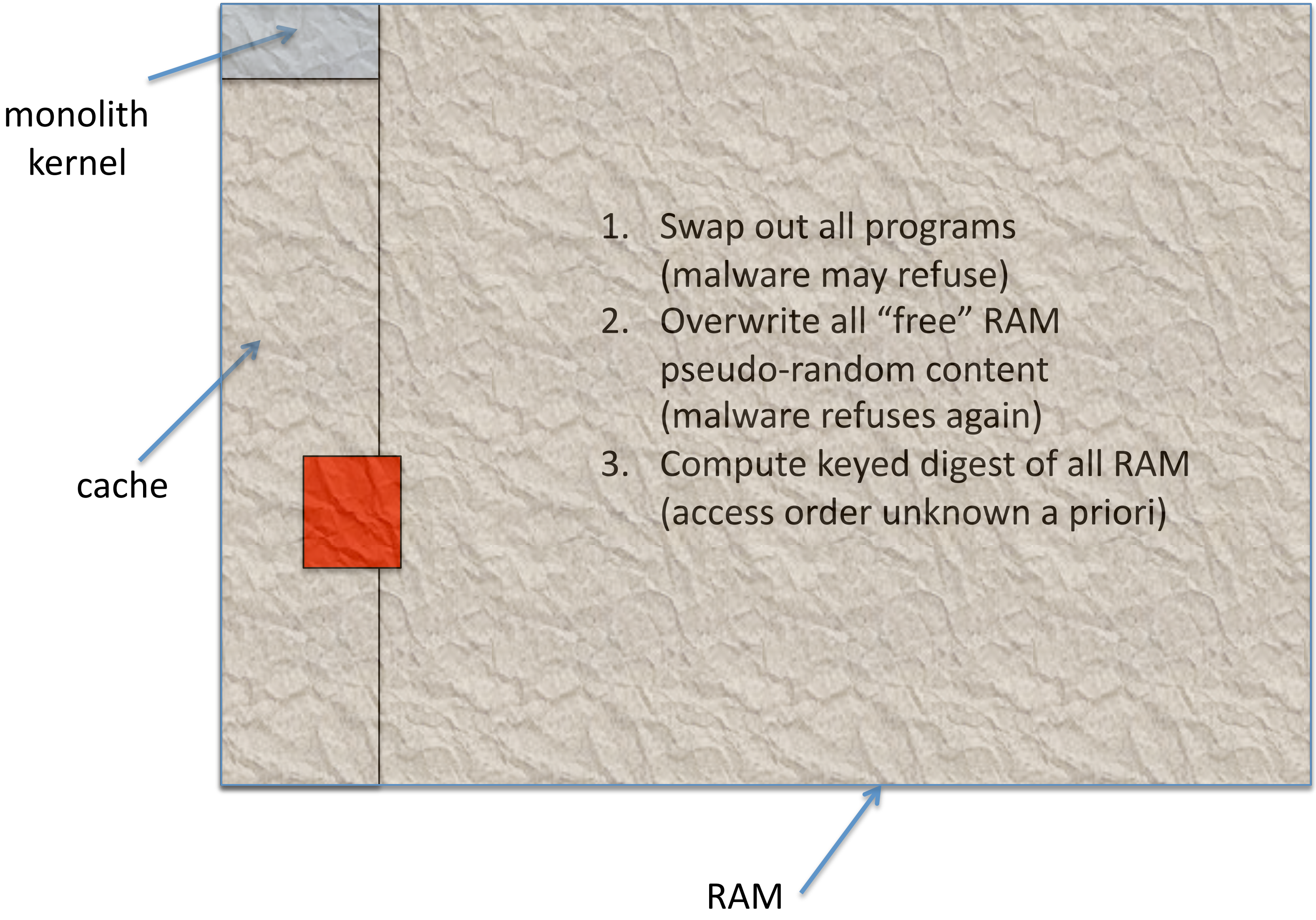
1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content(malware refuses again)

Contact markus@fatskunk.com for more details incl. improvements.



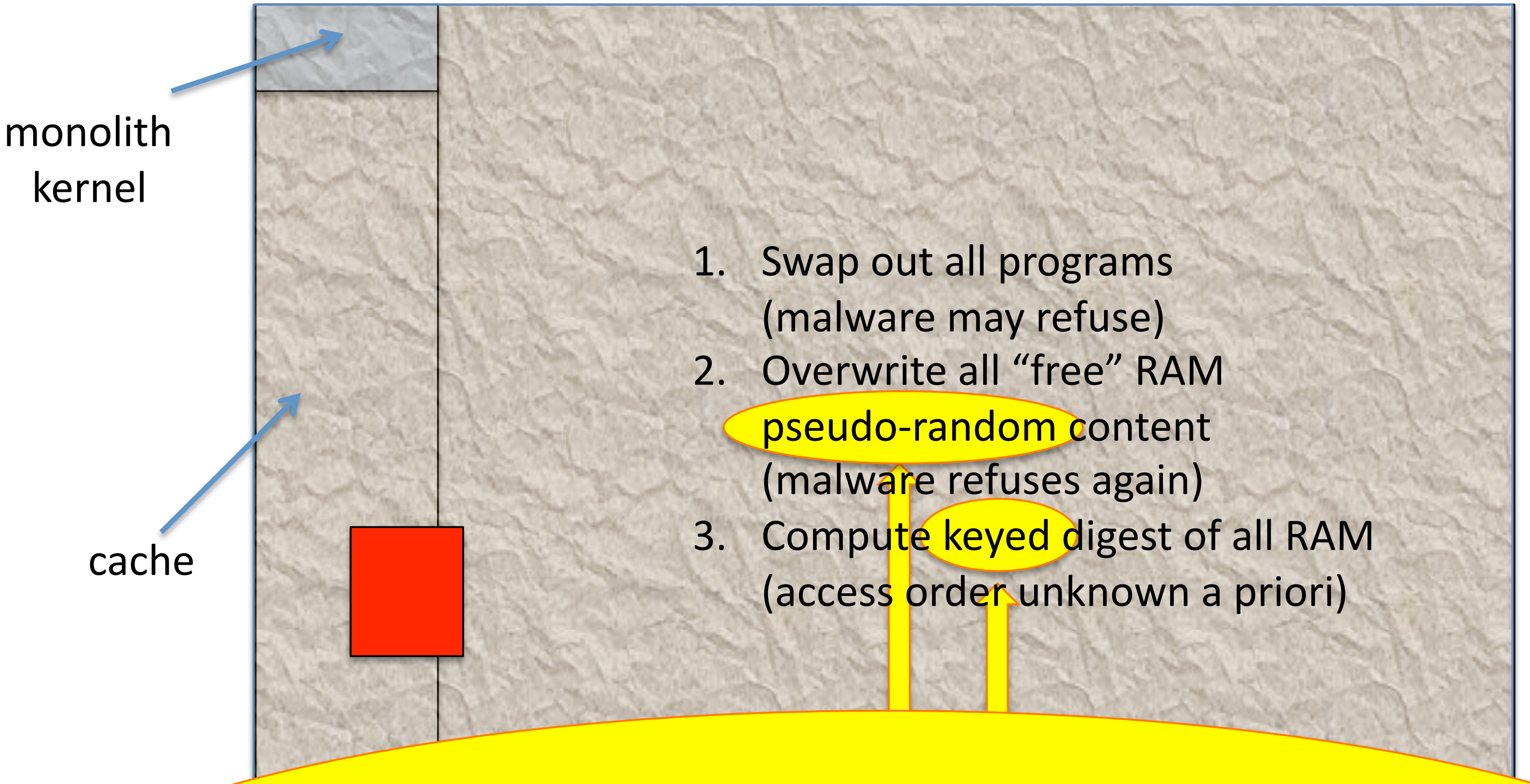
1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content (malware refuses again)
3. Compute keyed digest of all RAM (access order unknown a priori)

Contact markus@fatskunk.com for more details incl. improvements.



1. Swap out all programs
   (malware may refuse)
2. Overwrite all “free” RAM
   pseudo-random content
   (malware refuses again)
3. Compute keyed digest of all RAM
   (access order unknown a priori)

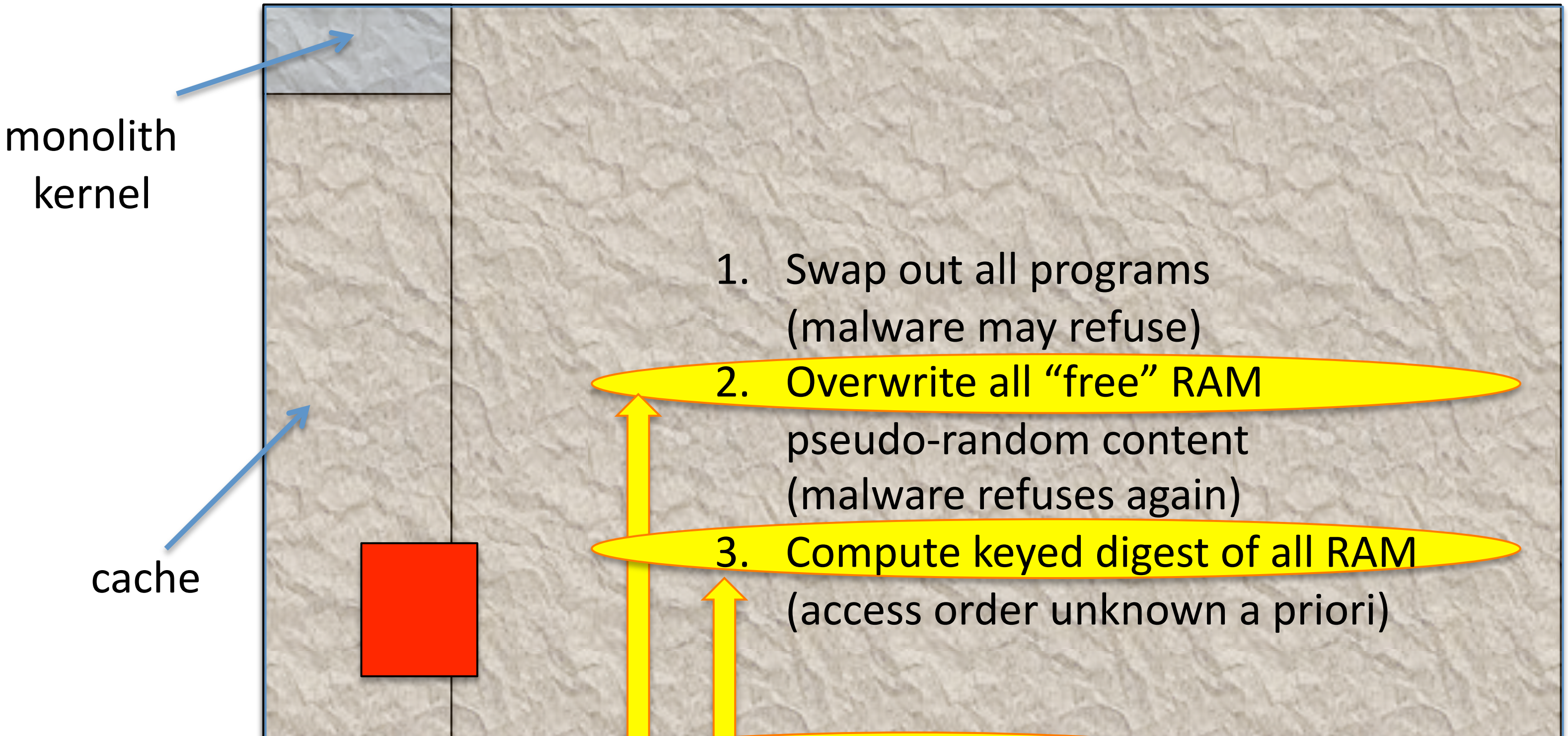
Contact markus@fatskunk.com for more details incl. improvements.



1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content (malware refuses again)
3. Compute keyed digest of all RAM (access order unknown a priori)

External verifier provides this

Contact markus@ provements.

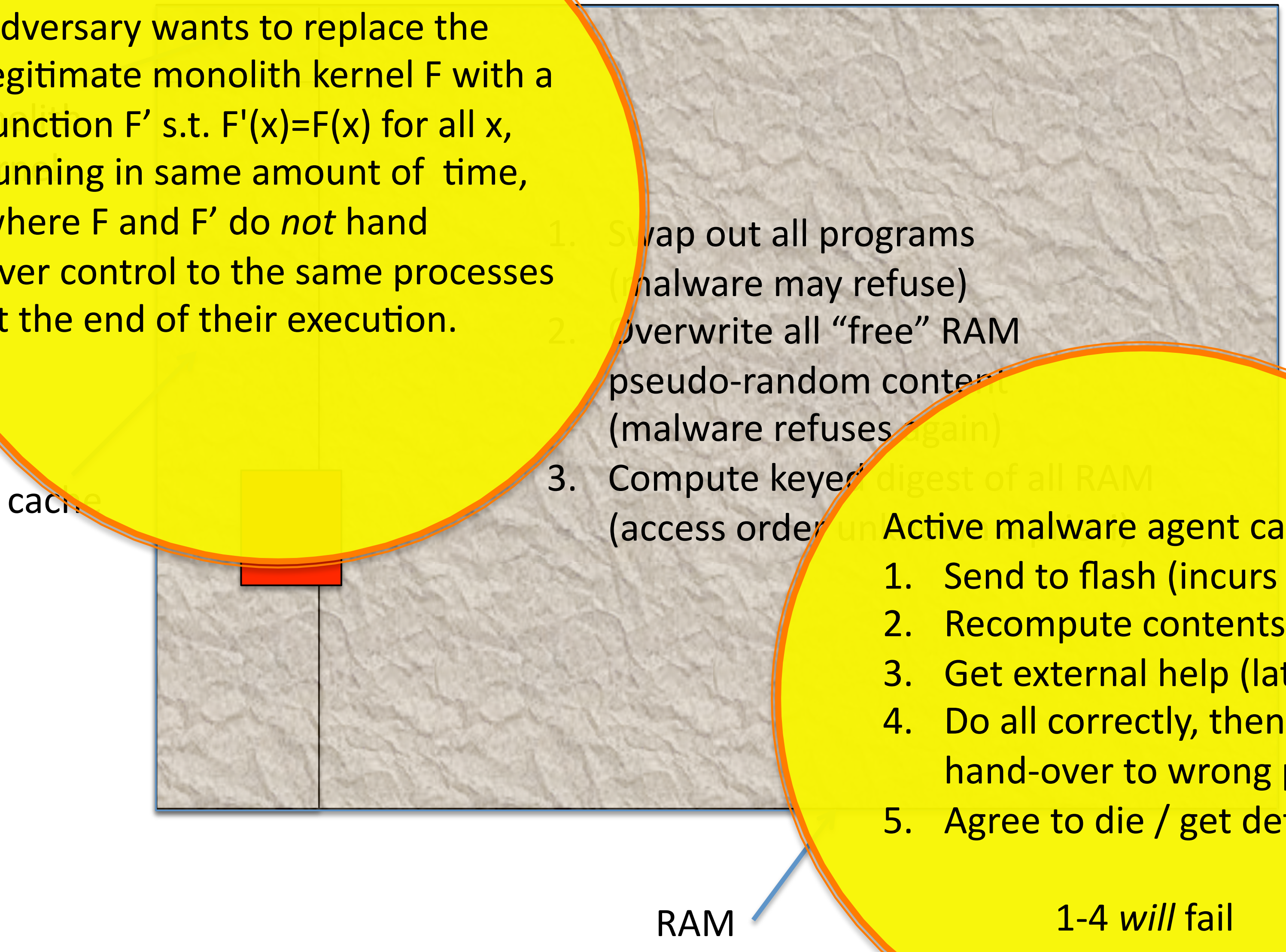


1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content (malware refuses again)
3. Compute keyed digest of all RAM (access order unknown a priori)

External verifier will time this
(and check result of computation)

Contact markus@ provements.

Adversary wants to replace the legitimate monolith kernel F with a function F’ s.t. F'(x)=F(x) for all x, running in same amount of time, where F and F’ do *not* hand over control to the same processes at the end of their execution.



1. Swap out all programs (malware may refuse)
2. Overwrite all “free” RAM pseudo-random content (malware refuses again)
3. Compute keyed digest of all RAM (access order unknown a priori)

Active malware agent can:

1. Send to flash (incurs delays)
2. Recompute contents (ow!)
3. Get external help (latency)
4. Do all correctly, then cause hand-over to wrong process
5. Agree to die / get detected

1-4 *will* fail

Contact markus@fatskunk.com for more details incl. improvements.

Some details

- Only requirement: know amount/type hardware
- Full use of caching (instruction + data)
- Strategy to maximize penalty for flash access
- Two adversarial models: external attacker or no
- SIM card can be used as low-latency timer

Contact markus@fatskunk.com for more details incl. improvements.

Some stats

- Variant implemented - takes <3s on 256MB, 600 MHz Android board
- Speedup for multi-core
- Detects *all* active malware – retroactively
- Provable security – no heuristics
- Suitable for mobile platforms
- Can be combined with a "secure rsync"

Contact markus@fatskunk.com for more details incl. improvements.