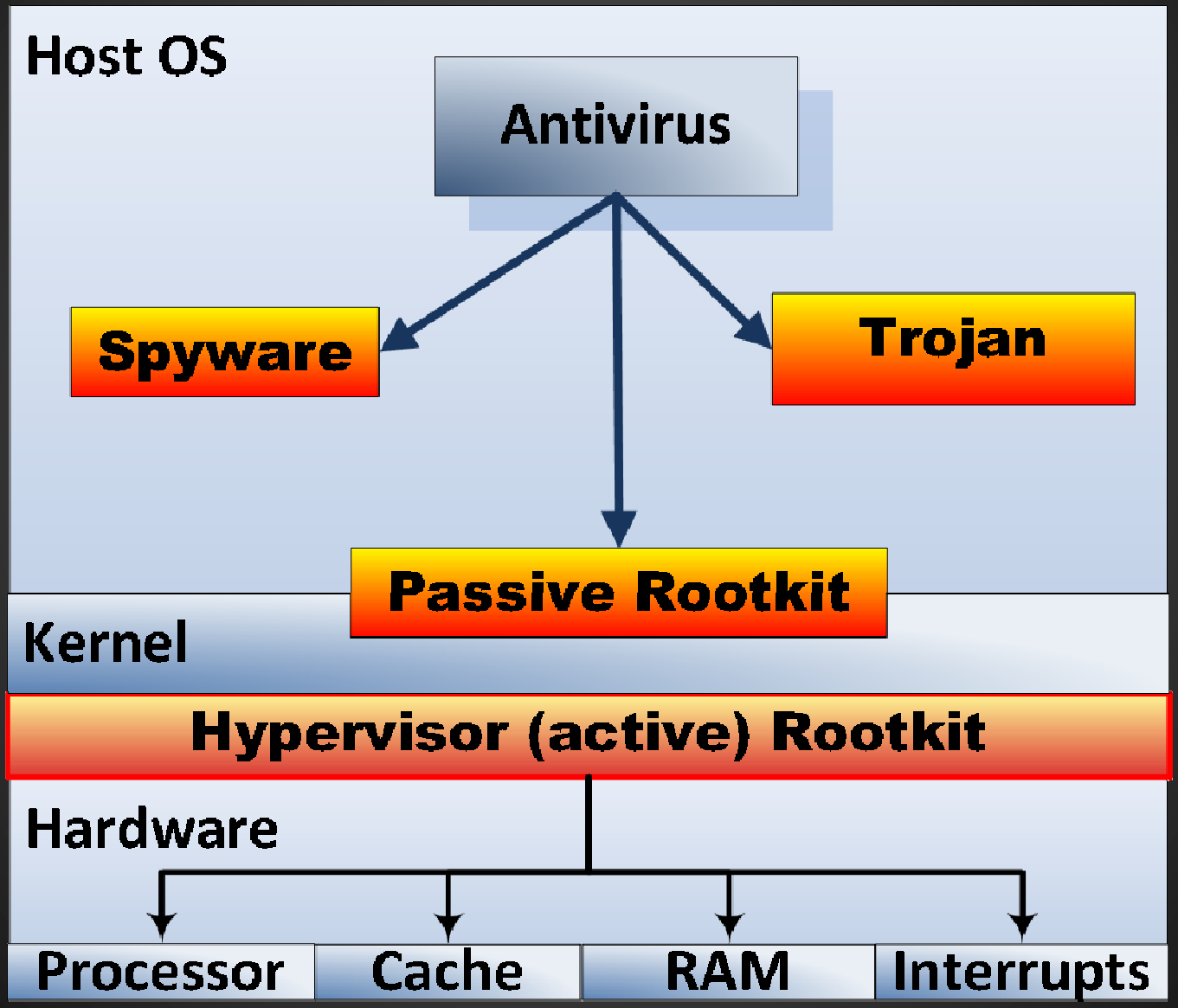
Mobile Malware: Why the traditional AV paradigm is doomed, and how to use physics to detect undesirable routines

Guy Stewart
VP Engineering
Fatskunk Inc.

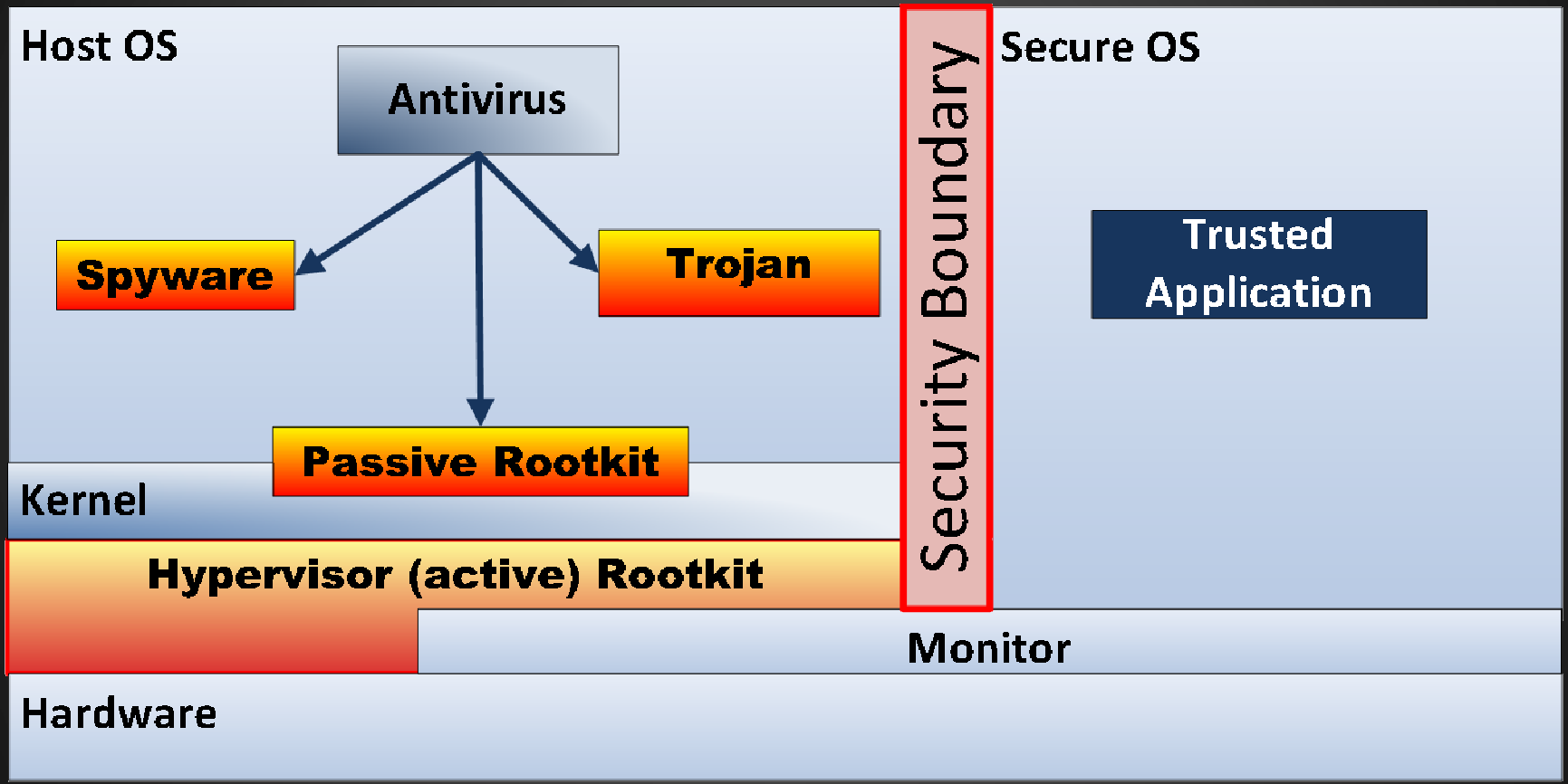
The Malware Problem

Trojans, Rootkits, the Zero Day Apocalypse

Threats


Host OS
Antivirus
Spyware
Trojan
Passive Rootkit
Kernel
Hypervisor (active) Rootkit
Hardware
Processor
Cache
RAM
Interrupts

Threats


Host OS
Antivirus
Spyware
Trojan
Passive Rootkit
Kernel
Hypervisor (active) Rootkit
Security Boundary
Secure OS
Trusted Application
Monitor
Hardware

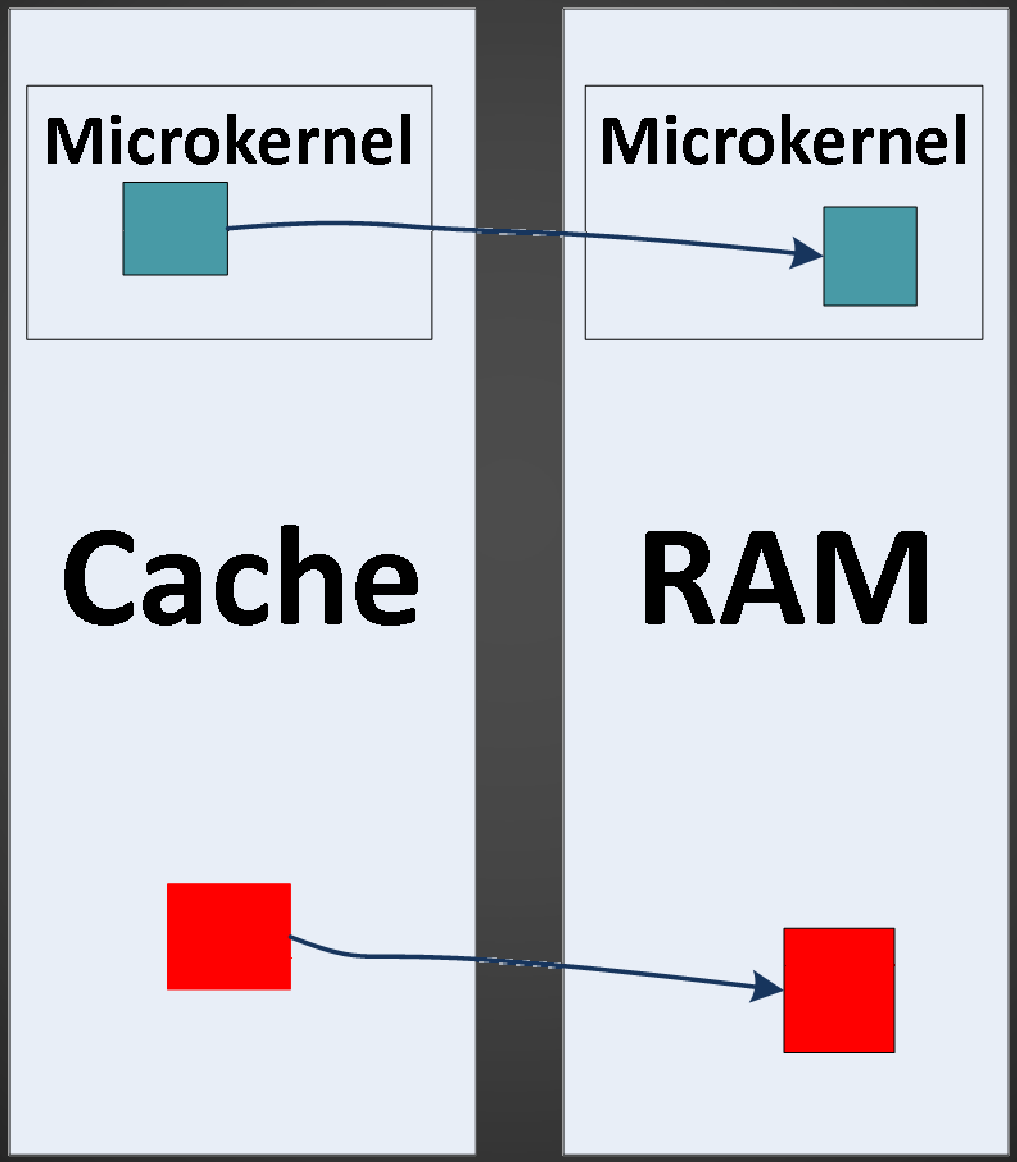
Untrustworthy Supply Chains

Software Attestation

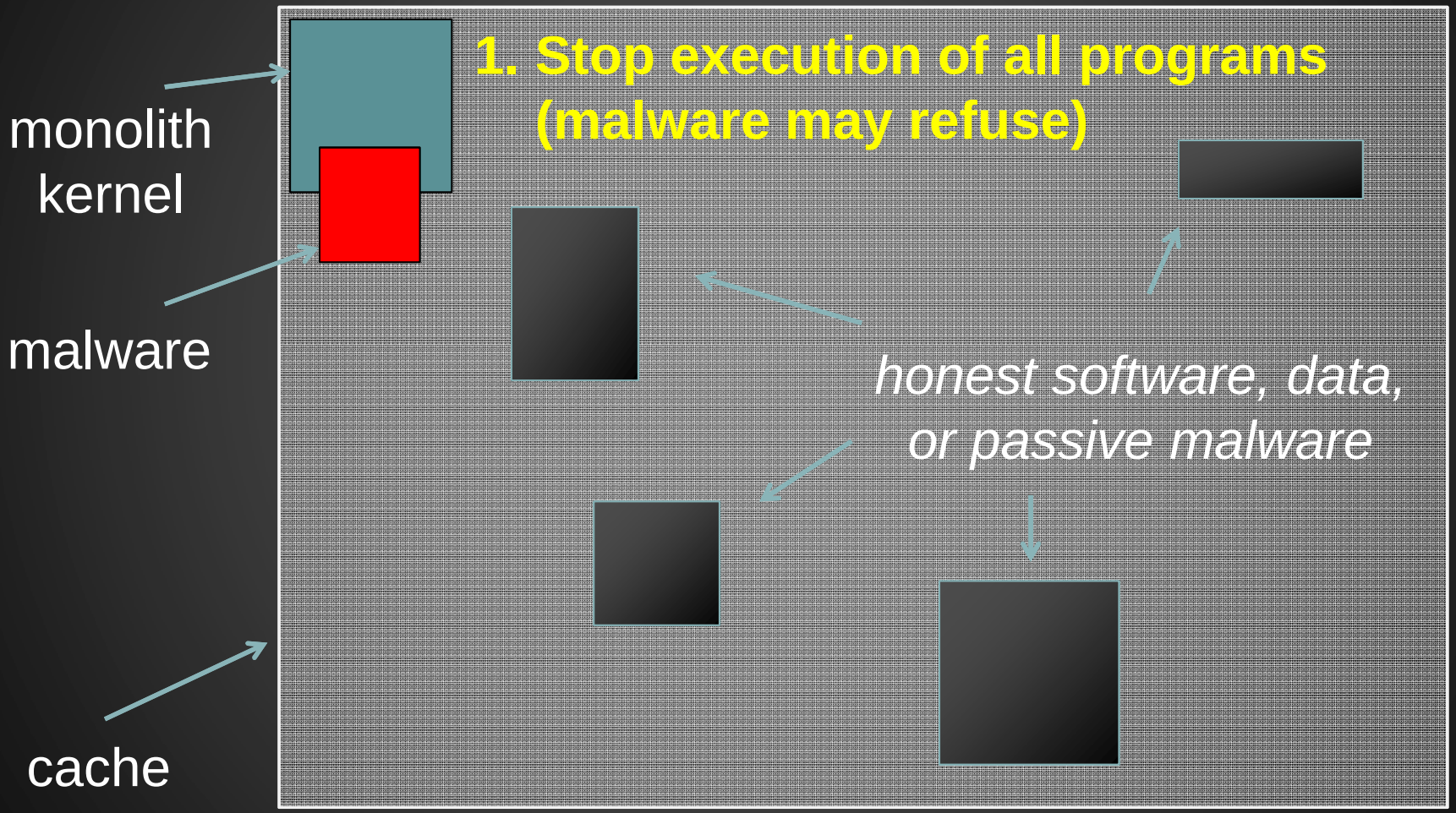
Introduction to Software Attestation using Principles of Physics

Approach: Measure by Displacement

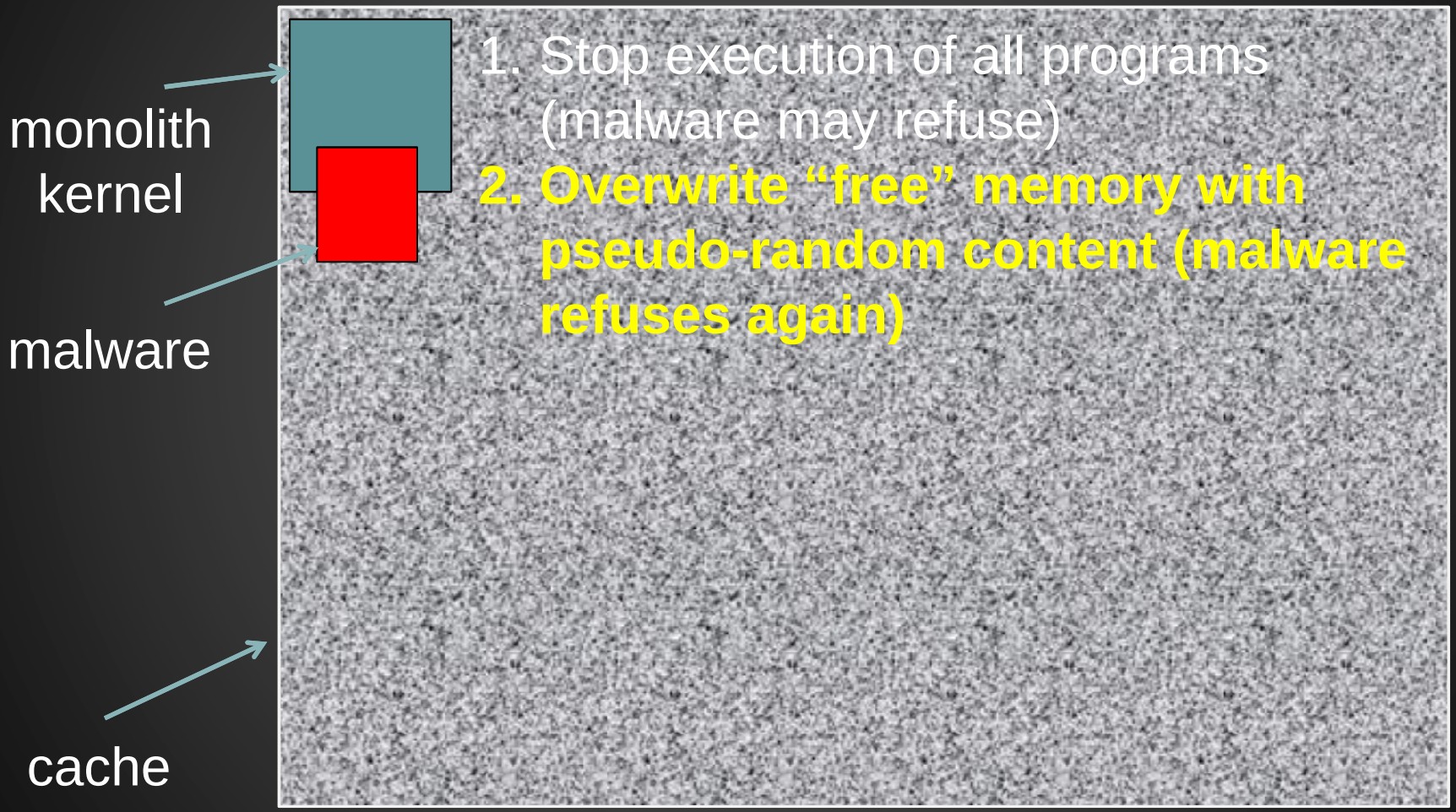
The software Space / Time trade-off



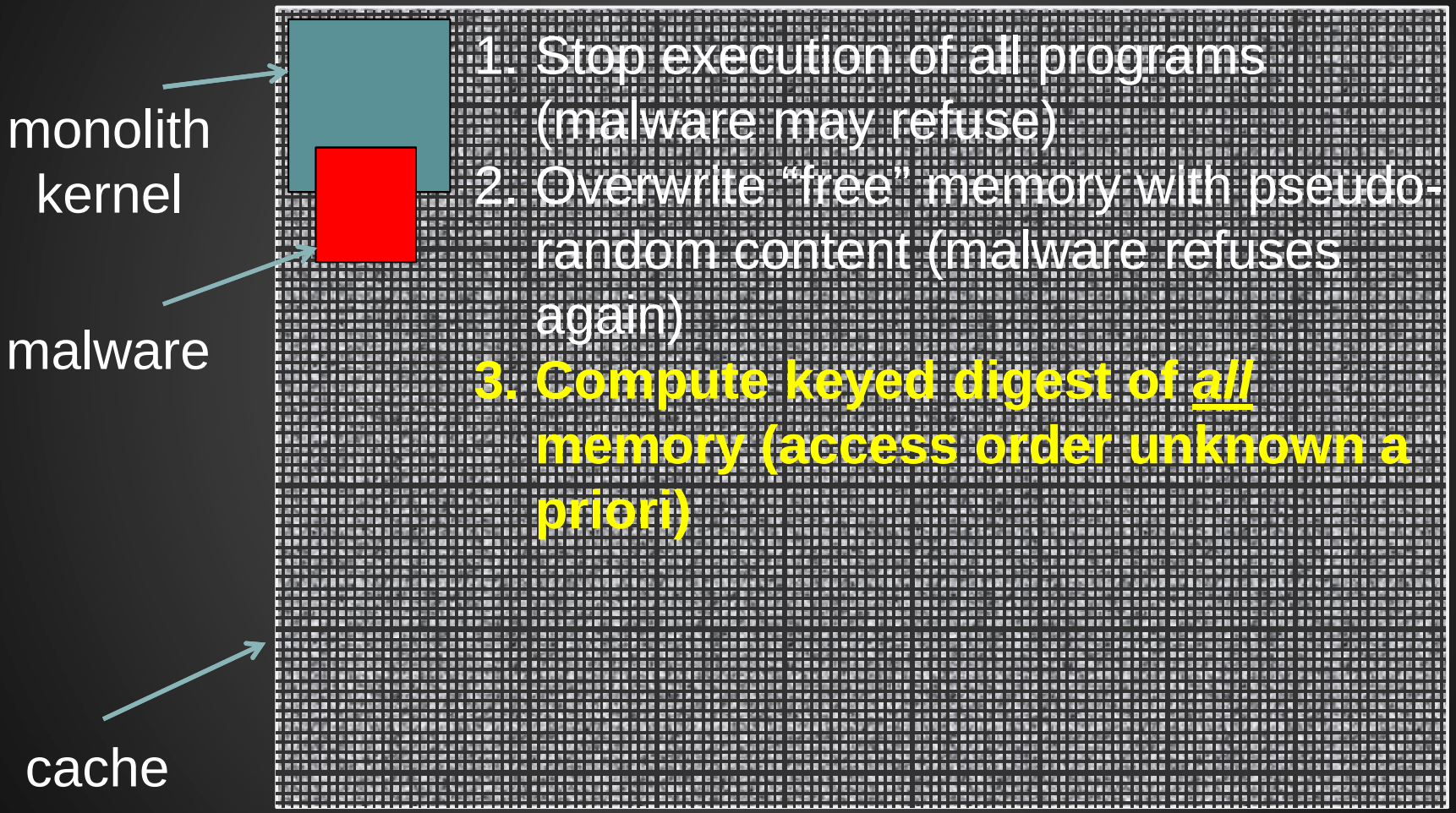
Approach


monolith kernel
malware
cache
1. Stop execution of all programs (malware may refuse)
honest software, data, or passive malware

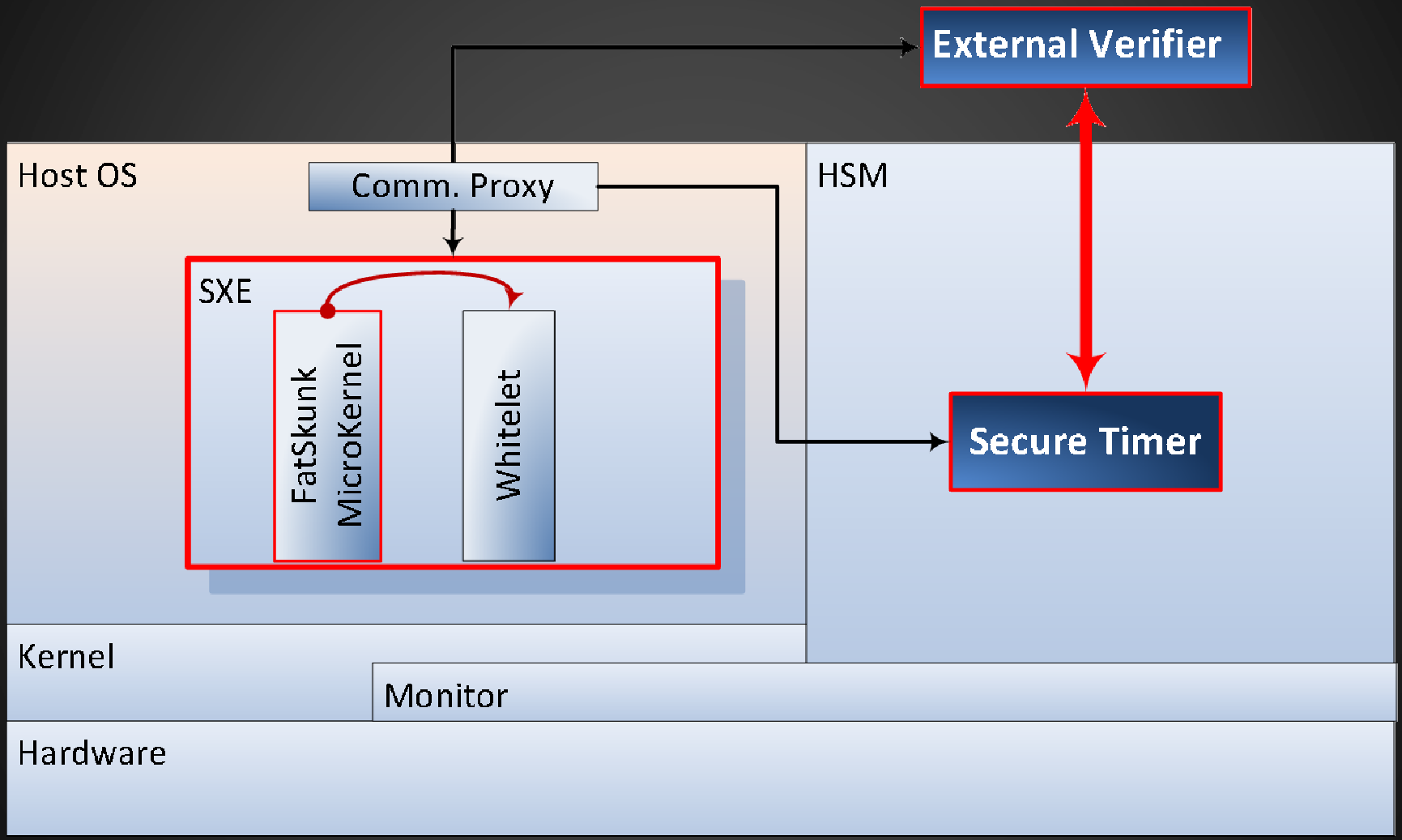
Approach


monolith kernel
malware
cache
1. Stop execution of all programs (malware may refuse)
2. Overwrite “free” memory with pseudo-random content (malware refuses again)

Approach

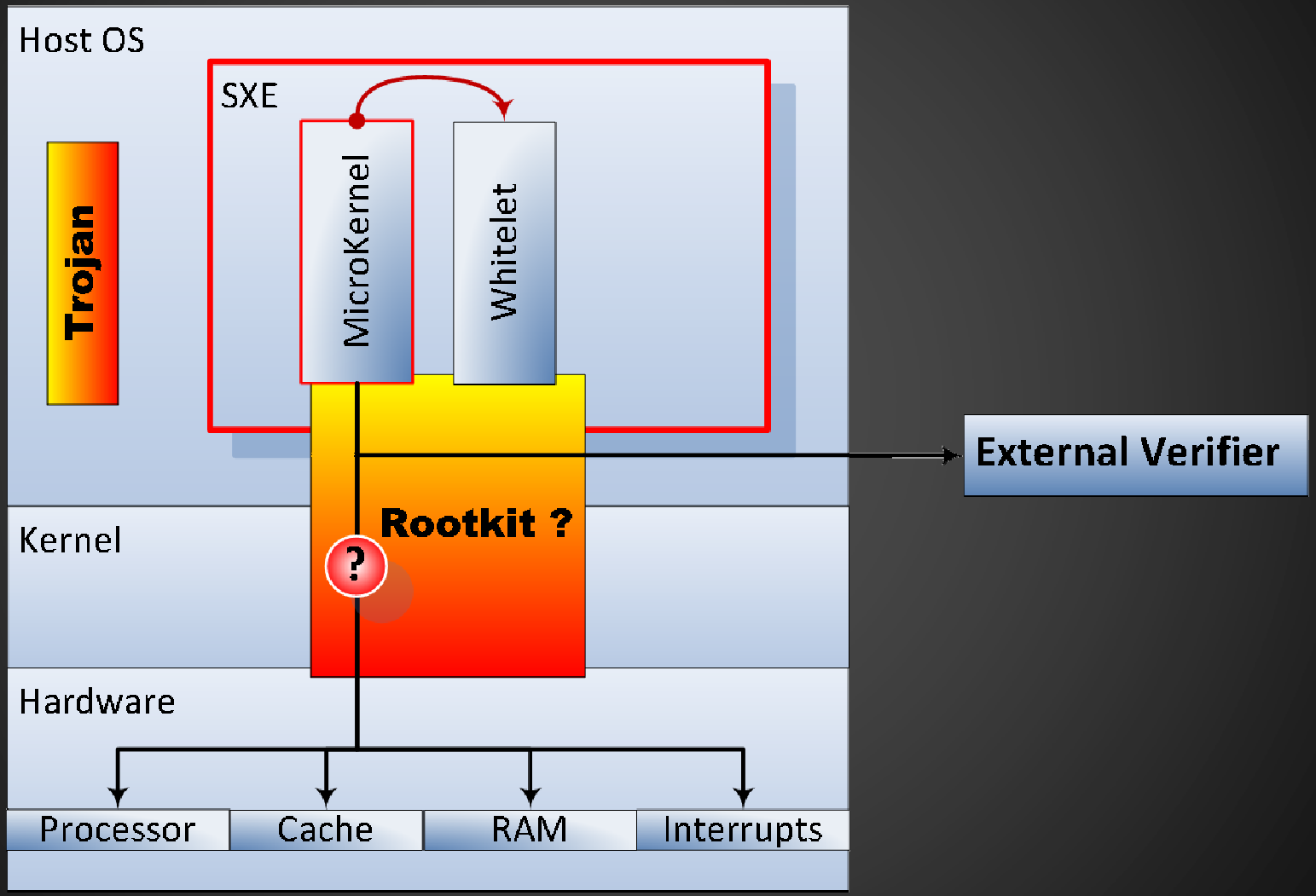

monolith kernel
malware
cache
1. Stop execution of all programs (malware may refuse)
2. Overwrite "free" memory with pseudo-random content (malware refuses again)
3. Compute keyed digest of all memory (access order unknown a priori)

Verify results

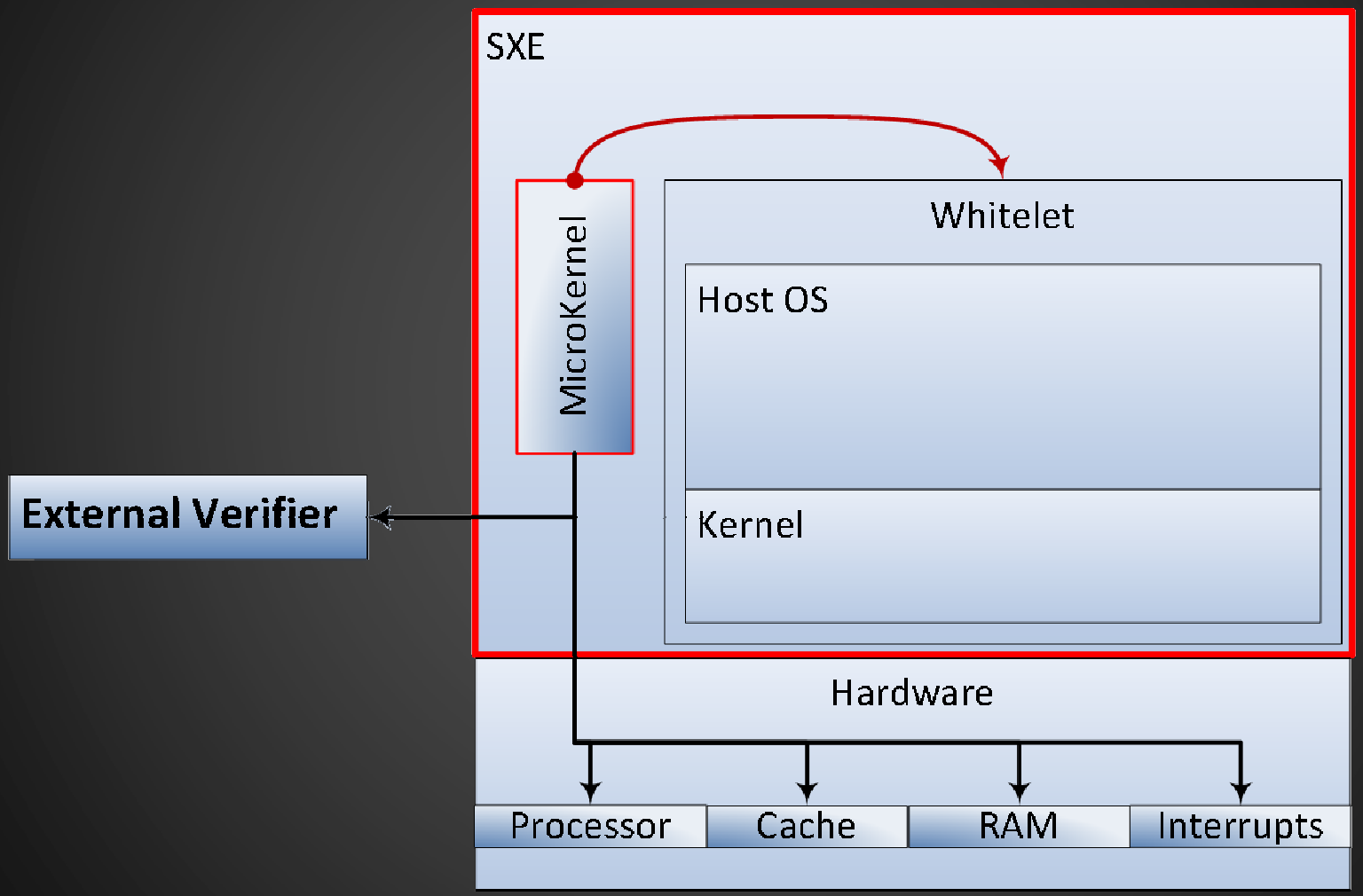


Commercial Applications

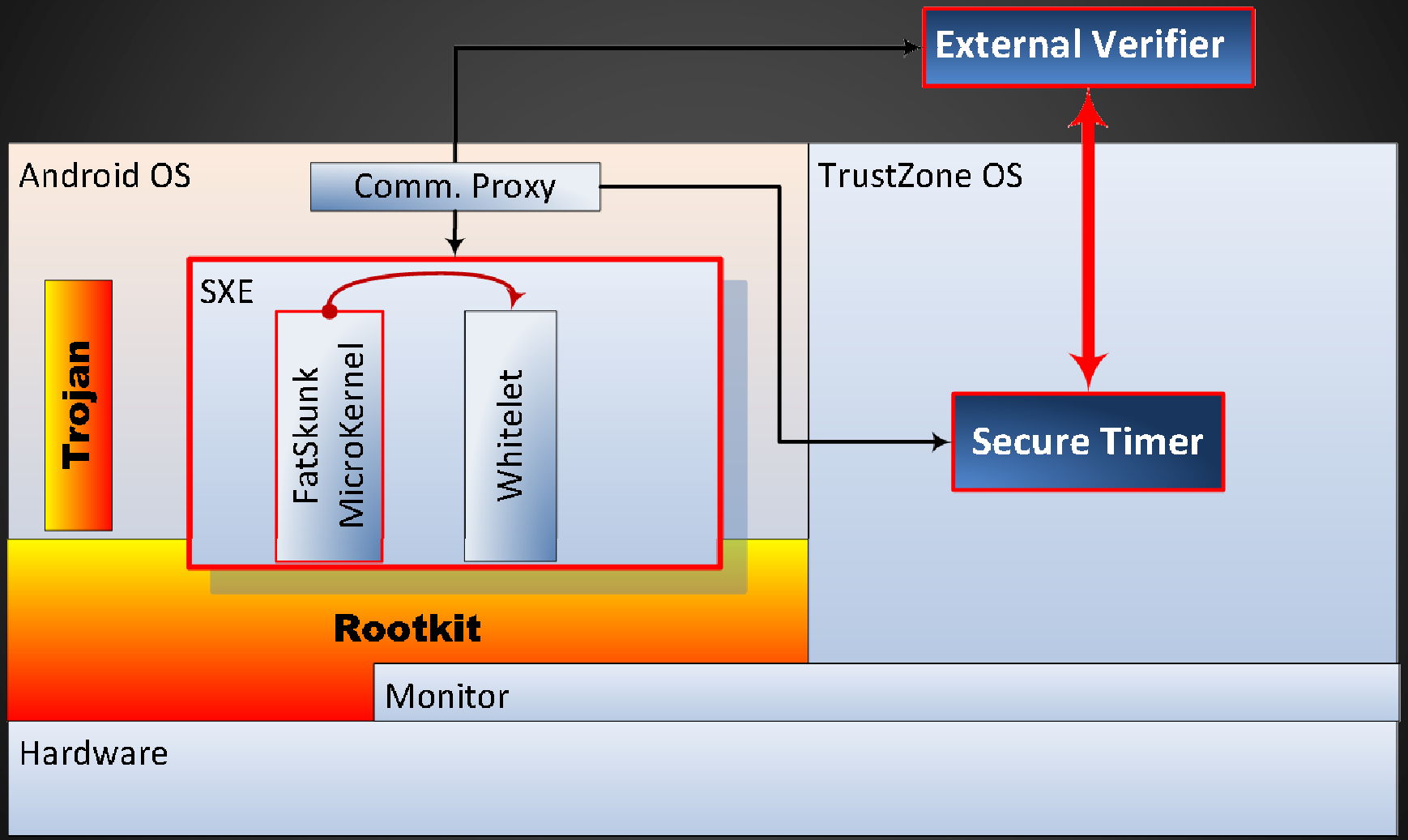
Secure Execution Environment (SXE)



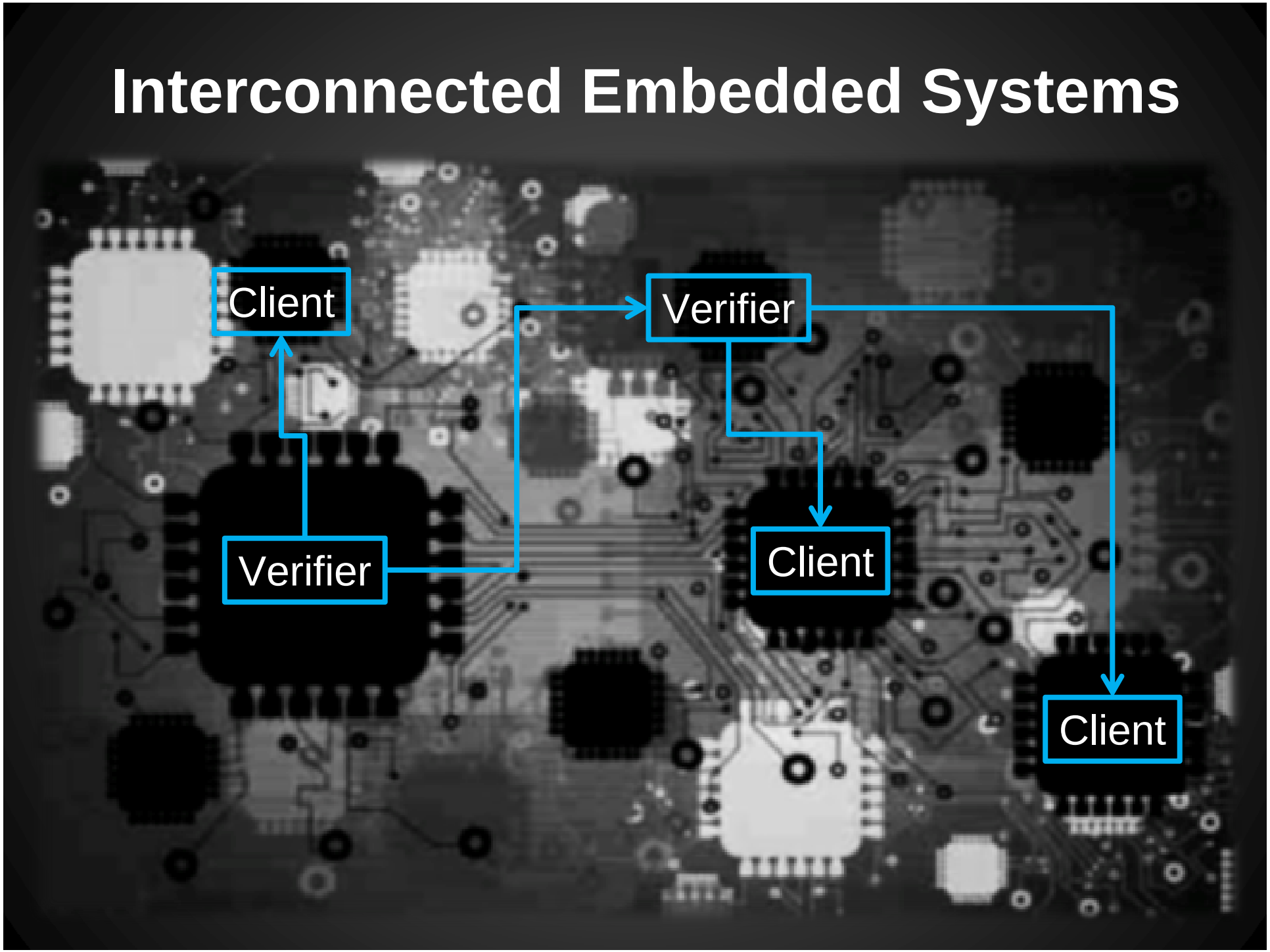
OS Secure Boot



TrustZone Normal World



Interconnected Embedded Systems



FatSkunk.com
Secure Execution Environment
The Mobile Malware Problem
Root of Trust
Secure Boot
Integrity
FatSkunk
Trojans
Malware
Rootkit
Jailbreaks
Untrustworthy Supply Chains
Interconnected Systems
Blacklisting
Software Based Attestation
Time and Space
Principles of Physics
Secure Data Store
External Verifier
Measuring Displacement
Microkernel
Guy Stewart : Guy@FatSkunk.com