RESEARCH ARTICLE OPEN ACCESS

Hypervisor Based Security

Anshumali Bhushan[1], Dr. Deepa Sharma[2]
[1,2]*Physics Department Himalayan Garhwal University Pauri, Uttrakhand, India*

Abstract:
Physical sensor-based hypervisor will be able to trap and detect hence forth undetected attacks / malware.
It is designed to fill the gap in cyber defense against APTs there by solving the long standing national level security gaps across endpoints, mobile phones, data centers and embedded systems.
Easily extended to multiple use cases, apart from the kernel and application protection that is applicable to endpoint devices, mobile devices and data center entities.
Hypervisor has a technology which has an upper hand over the old antivirus techniques which used sandbox method for prevention against Malware attack.
Key protections offered by Hypervisor will include prevention against malicious kernel code injection and approval of code execution in kernel mode only

Date of Submission: 30-01-2020 Date Of Acceptance: 15-02-2020

I. INTRODUCTION

Recent years have been memorable for cybercrimes against high-profile target, it included major attacks on defense establishments, corporates, social networking sites, movie-streaming giants, music services etc. Conventional security mechanisms are not effective against the advanced form of attacks that are frequently employed against these high value targets. We are designing a hardware enforced security solution which can be further enhanced to counter advanced threats there by filling the gap in nation's defense against cyber attacks of today.

II. CYBER ATTACKS OF TODAY

Overtime, hacking has shifted its intentions as well as its sophistication. Originally created for thrills to test one's capability to gain access to protected areas, the focus inevitably moved to money and in recent times, the attacks have become even more brutal with state sponsored teams, organized crime rings, targeting defense establishments and large corporations. A standout feature of many of these attacks are the complexity of the malware involved making conventional security solutions are woefully inadequate to detect or prevent the same.

Threats that avoid detection and harvest valuable information over a long time are known as Advanced Persistent Threats (APTs). Traditional security measures such as antivirus, firewalls etc. cannot provide protection against APTs thereby leaving systems vulnerable to data breaches. The consequence of an APT attack is devastating as the attack may continue for a long time uninterrupted due to the limitations of most of the security solutions out there.

A. State Sponsored Attacks(Critical Infrastructure)

State sponsored espionage incidents on critical infrastructure are continuing to rise and these threat actors have objectives aligned with political, commercial or military interests of their country of origin. One of the most famous of these types of attack is 'Stuxnet' which targets SCADA systems and is believed to have caused substantial damage to Iran's nuclear program. Recently India woke up to the news of a breach in Kundankulam nuclear power plant which is suspected to be the handiwork of Lazarus group from North Korea.

One of the goals of state sponsored attacks is to remain persistent for months to years by not making noise and they achieve this by having APT capabilities in the malware.

B. Organized Crime Rings (https://www.happiestminds.com/infographics/advanced-persistent-threats.pdf)

Organized crime rings are known to use APTs in the effort to gain personal financial information, intellectual properties etc. from corporates. It is estimated that more than 1 BN $ have been stolen from over 100 financial institutions by the Carbanak cyber gang.

C. Anatomy of APT

Sophisticated and systematic attacks where the intruder establishes long term presence in the system are called Advanced Persistent Threats

(APTs) Security researchers have found that, many of the APTs have kernel mode or even firmware component that shields the malicious code from getting detected thereby ensuring that the attack continuous uninterrupted. Most of the security solutions of today are not a match against these highly evolved attack vectors.

1) Kernel Mode Rootkit

A kernel mode root kit runs at the same privilege level as the OS kernel and hence are hardest to detect and clean. Kernel mode rootkits can manipulate the kernel, memory and other system elements. Primary job of such rootkits is

- Disable security measures such as antivirus
- Conceal malware

Rootkits conceal other malware and malicious payloads until the time is right for the attack, that is why rootkits are a preferred tool in stealthy threats like Stuxnet, Turla etc. Often the attacker applies creativity in building the rootkit and then leverages off-the-shelf malwares for rest of the crime. There are also instances where kernel mode rootkits not only hide the presence of malicious user mode components but also leverages the kernel mode privileges to perform sophisticated attacks such as injecting arbitrary code to running processes, directly in the context of kernel. Below are details of how Turla made use of kernel mode privileges to carry out a highly sophisticated attack on Windows.

a) Turla APT

Turla is an advanced APT and is suspected to be state sponsored. Some of the key functionality of Turla is implemented as a kernel driver and this allows the malware to bypass the in-built security features in Windows kernel.

Turla kernel driver is not 'signed' and hence should not be loaded by Windows, the malware authors negate this using a legit signed Virtual Box driver that has a known vulnerability, by exploiting this vulnerability, the driver-signature verification is turned off and malicious kernel driver is loaded. The malicious kernel driver tampers with certain kernel routines there by making PatchGuard ineffective. The modification of kernel routines are possible as the kernel driver is executing at the same privilege level as the Windows kernel. The malicious kernel driver then proceeds to hook a number of system calls mainly to hide/protect its user-mode components.

2) Firmware and Hardware Attacks

Once considered as urban legends, recent changes in threat landscape have proved that firmware and hardware attacks are a reality. UEFI rootkits that once were known to exist as proofs of concepts have been now discovered in the wild, deployed by Sednit group suspected to be targeting government organizations in the Balkans as well as Central and Eastern Europe.

D. Limitations of Conventional Solutions against APTs

With its kernel mode privileges, APTs such as Turla are able to hide the user mode malicious components from virus scanners, it also defeats conventional sandbox techniques by disabling any in-host hooking mechanisms. Turla is a classic example of APT that renders conventional security mechanisms completely ineffective.

E. Limitations of Signature based detection

Conventional security products work by computing the digital signature of each object and comparing the same with a database of known malicious signatures, this is an effective method as long as the signature exists in the database and the malicious object is visible to the tool for computing the signature for comparison.

Due to the reliance on known signatures, these technologies will fail to detect zero day attacks or any malware, signature of which is not present in the database. Malware authors also alter signatures of existing malicious code to avoid detection using techniques such as

- Code permutation
- Register renaming
- Expanding and shrinking code
- Insertion of garbage code or other constructs

According to Trend Micro, bad actors create a million new malicious objects every day. Some of these are truly new threats, but most are variations on existing malware. Unfortunately, it can be several days after a new malicious object appears in the wild before security vendors update their signatures (although it's not unusual for two weeks to pass before a security vendor makes a signature available). Until the new signature arrives, conventional security controls will not detect the malware and

Organizations are vulnerable during that time.

Below declarations by Antivirus makers themselves may appear sensational but as far as countering advanced threats are considered it is definitely true.

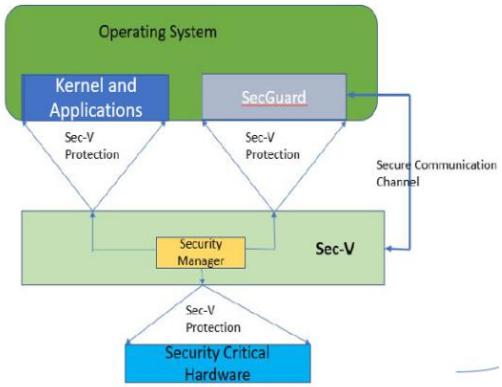
1) Limitations of Conventional Sandbox Techniques

Acknowledging the in-adequacy of signature-based technologies against advanced malwares, security vendors started embracing sandbox technologies. Sandbox techniques rely on behavioral analysis rather than signatures, sandbox simulate a network environment and hopes to fool the malicious object to demonstrate its true color.

The sandbox method was once effective, but malware have evolved to evade sandboxes too. Sandbox technologies typically use Virtual Machine environments but the VM environment inserts artifacts that allow advanced malware to discover that it is running in a virtual environment and will lay dormant there by evading detection.

III. HYPERVISOR - HARDWARE ENFORCED SECURITY SOLUTION AGAINST APTS

Its our assumption and premise that Hardware Enforced Security solution which can utilize advanced state of the art sensors utilizing physics-based data models, can fill the gap in India's cyber defense against APTs there by solving the long-standing national level security gaps across endpoints, mobile phones, data centers and embedded systems.



The key components in Hypervisor are detailed below.

A. User Interface

We also need to design a User Interface for alerting, policy configurations, secure update etc. This will provide RESTful APIs for configuration and management purposes.

B. The Security Hypervisor

Hence our Physics based data models will be a purpose-built security hypervisor targeted to detect, contain and analyze kernel mode APTs. The surge in APTs with kernel mode components.

1) Design Details

While designing Hypervisor, we need to ensure that it leverages hardware virtualization capabilities provided by hardware to create necessary isolation and protection against APTs. The Physics based sensors would be critical and highly accurate so that false alarms are minimized. Further Secure coding practices should be used during the development of Hypervisor, also care is taken to ensure that the total footprint of Hypervisor makes it amenable to formal verification.

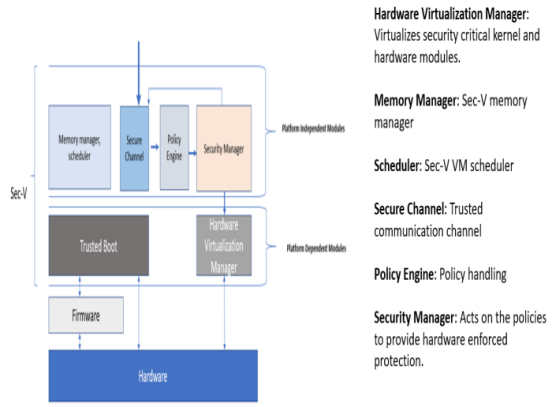
We also want to ensure that our Hypervisor is designed to be easily extended to multiple use cases, apart from the kernel and application protection that is applicable to endpoint devices, mobile devices and data center entities (servers, white box switches/routers etc.) It should also be tuned to work as a separation kernel as well as malware reverse analysis engine.

1) Proposed Booting Model

During system boot, the Hypervisor should be up after BIOS and must boot the OS to be protected in a virtual machine. This enables hypervisor to set desired security policies on the OS and detects malicious actions performed by malware, even the ones with kernel privileges.

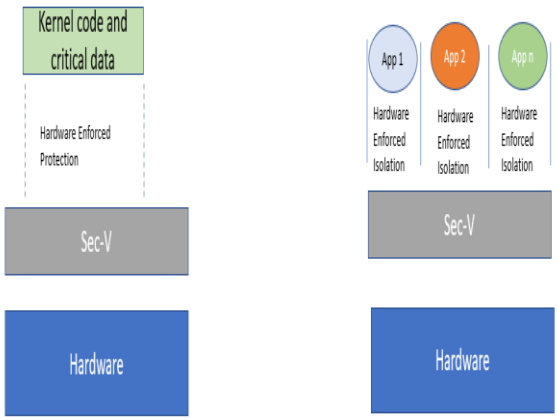
Hypervisor ensures that the OS under protection does not suffer from any performance

issues by leveraging hardware virtualization capabilities and selective virtualization of security critical components. By executing at a privilege level greater than that of the OS kernel, Hypervisor is in a unique position to identify attacks from kernel or IO devices.



2) Hypervisor for Kernel and Application Protection

A key functionality of Hypervisor is to provide kernel and application protection by providing the necessary isolation via hardware extensions. The level of protection offered is tunable via User Interface component. As the isolation and protection is via hardware, it prevents malware from breaking out or even go hiding there by enabling users to perform reverse analysis.

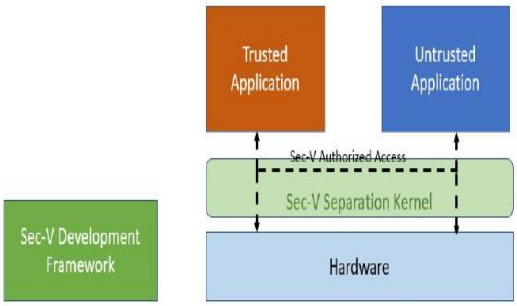


Some of the key protections that shall be offered by Hypervisor are

- Prevents malicious kernel code injection
- Permits only approved code execution in kernel mode
- Protects tampering of critical kernel data structures and security critical CPU registers.
- Permits only approved processes from executing
- Permits only approved user processes from gaining root privileges
- Prevents malicious applications from accessing unauthorized resources
- Prevents DMA attacks.

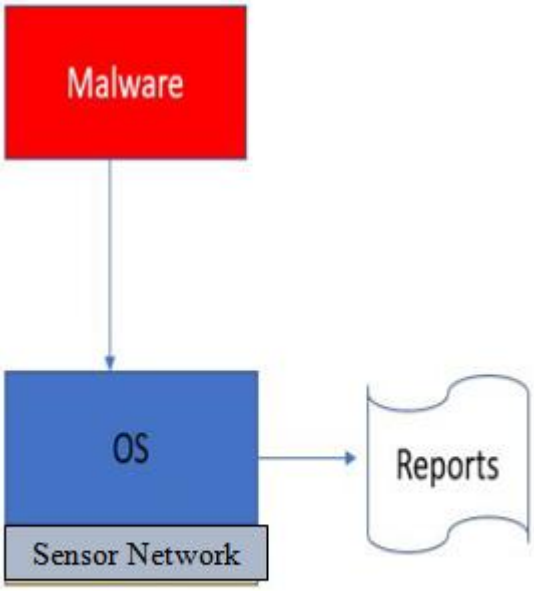
C. Hypervisor as a Separation Kernel

Hypervisor can easily be extended to function as a separation kernel there by providing isolation between multiple VMs and containers. The development framework provided as part of internal security suites that would enable developers to build independent secure software components.



D. Hypervisor for Malware Analysis

Hypervisor malware analysis mode will enable a user to perform deep malware analysis that is beyond the abilities of conventional reverse analysis tools. There will be no artifacts that would wan the malware about the presence of Hypervisor. Both static and dynamic analysis are supported and detailed reports of the executed behavior of the sample will be provided.



Malware: Malicious code

IV. CONCLUSION

The constantly evolving threat landscape where state sponsored attacks, corporate espionage etc. are becoming a norm rather than exception and imported security tools itself suspected to steal information, nations have started building advanced indigenous capabilities for defensive and offensive purposes. India has not pursued this approach and this oversight has resulted in a strategic gap in the Operating System level security percolating down to the kernel level operations and hardware. Our product enabled with deep understanding of kernel, hardware and contemporary security has come up with a hardware enforced security solution that can be the answer to the holes in India's national security.