usenix

THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

Bitdefender

Hypervisor memory introspection at the next level

User-mode introspection and protection of live VMs

Andrei LUȚAȘ

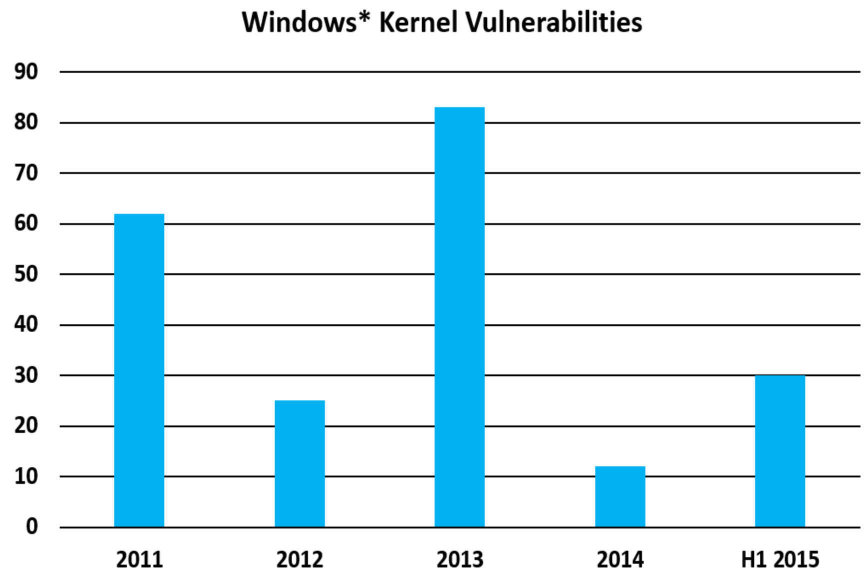
Senior Introspection Research Lead, Bitdefender

2015 USENIX Annual Technical Conference

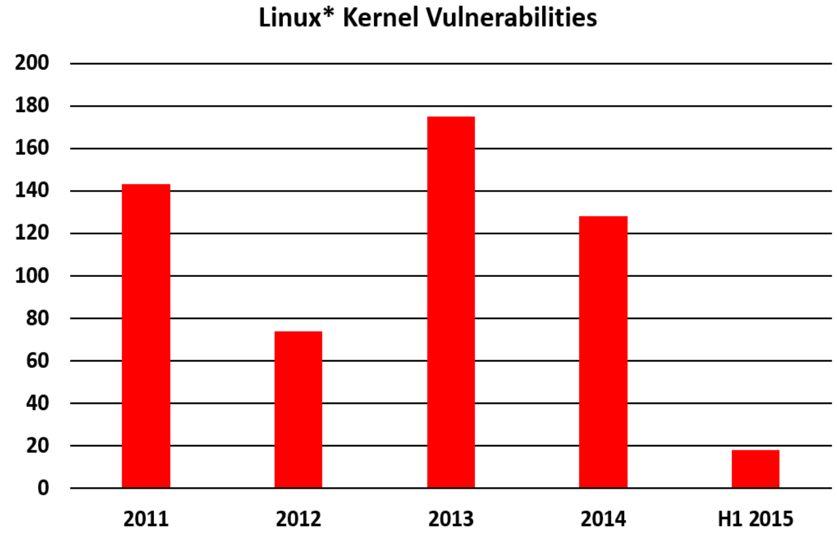
Santa Clara, 8-10 July, 2015

Security issues we are facing today

- Advanced malware types
  - Rootkits
  - Kernel exploits
  - Zero-days
- APTs, botnets, cyber-espionage etc. heavily rely on those…
  - CVE-2012-0158 → APT28
  - CVE-2013-1347 → Energetic Bear
  - …

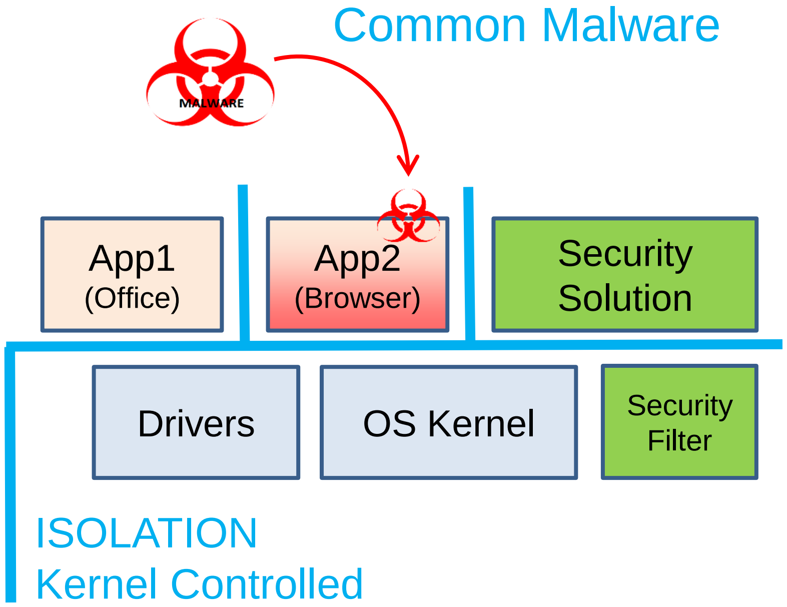


source: based on nvd.nist.gov



source: based on nvd.nist.gov

The lack-of-isolation problem



Common Malware

App1
(Office)

App2
(Browser)

Security Solution

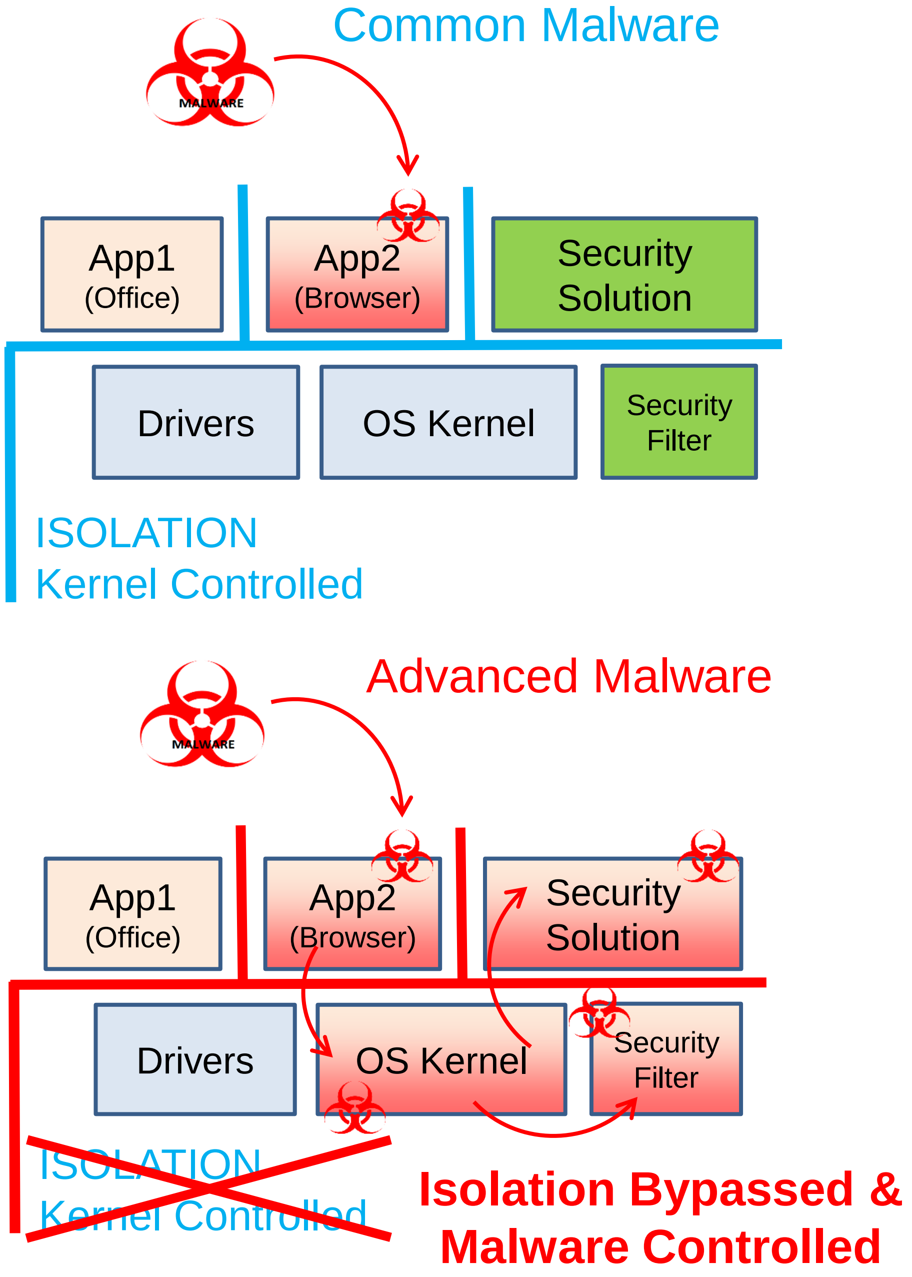
Drivers

OS Kernel

Security Filter

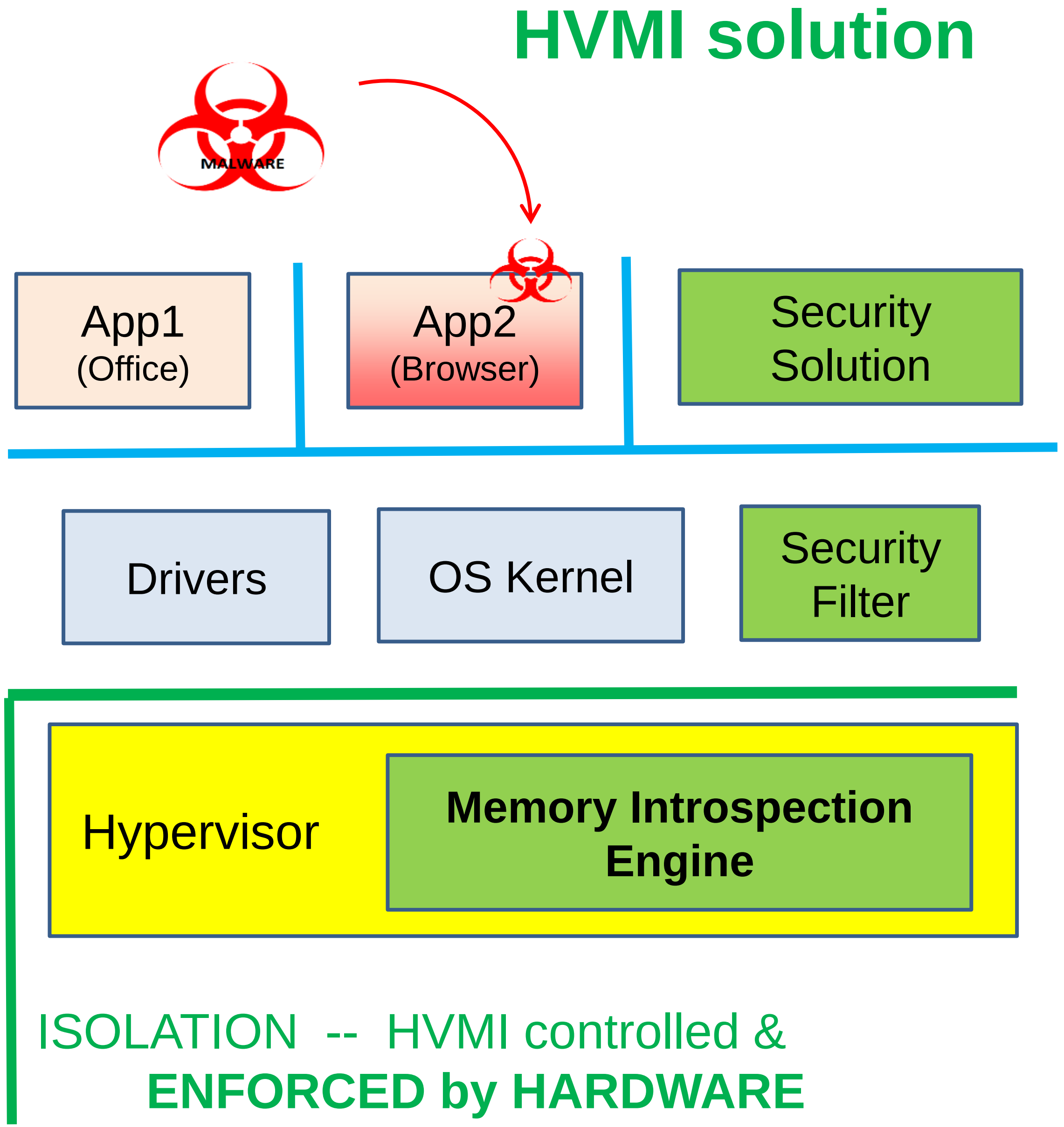
ISOLATION
Kernel Controlled

The lack-of-isolation problem

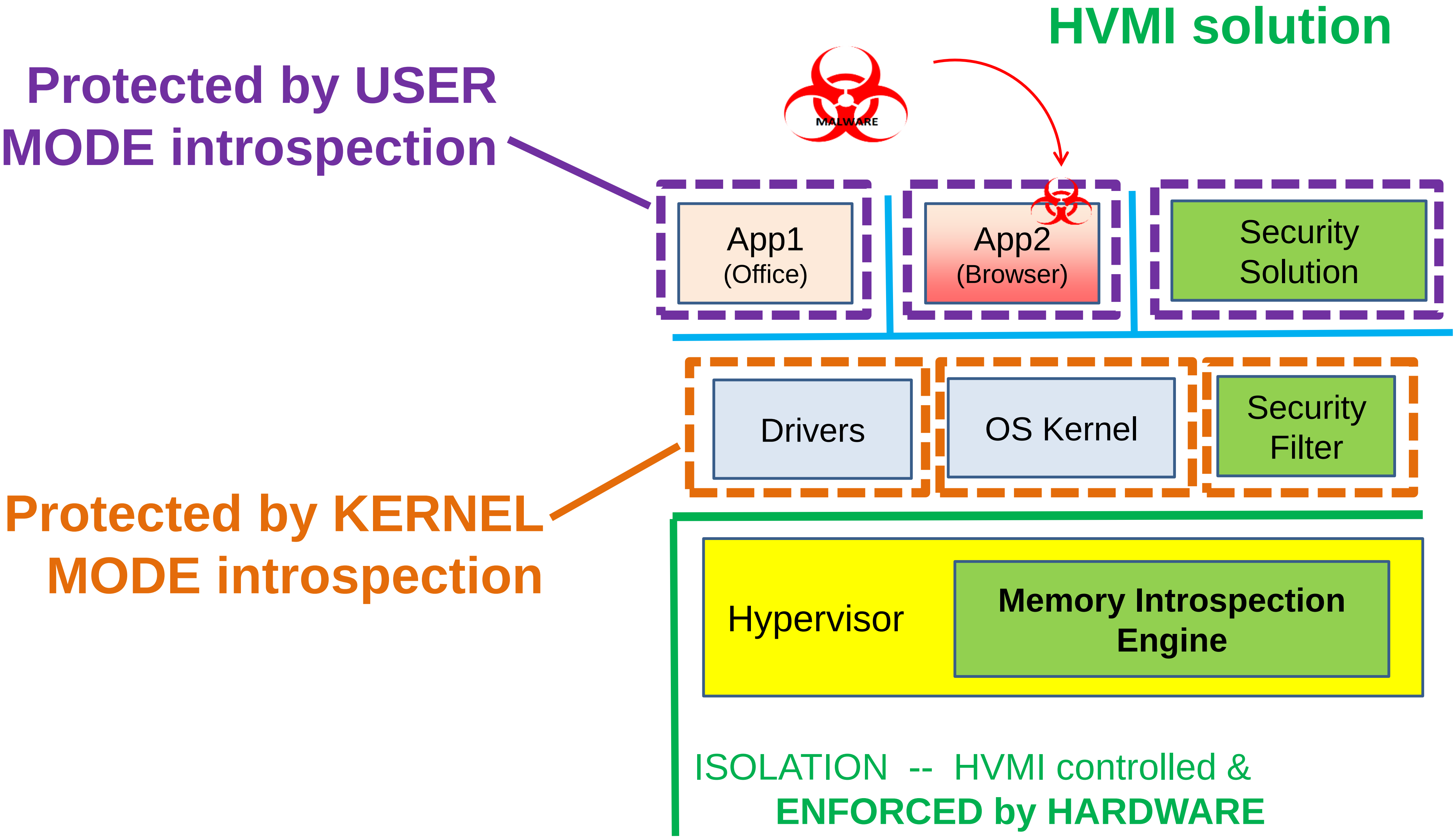


Isolation Bypassed &
Malware Controlled

Solving the lack-of-isolation problem



Solving the lack-of-isolation problem


HVMI solution
Protected by USER MODE introspection
MALWARE
App1 (Office)
App2 (Browser)
Security Solution
Drivers
OS Kernel
Security Filter
Protected by KERNEL MODE introspection
Hypervisor
Memory Introspection Engine
ISOLATION -- HVMI controlled & ENFORCED by HARDWARE

What is memory introspection?

- **provide security from outside the guest OS**
  - not relying on guest OS – can be compromised by advanced threats
  - relying on hardware accelerated virtualization (Intel* VT-x, EPT, …)
- analyze raw memory image of guest OS and applications
  - hook / mark 4K pages as **non-execute** or **non-writable**
- audit access of those areas by the code running in VM
  - write attempts, execute attempts
  - allow or deny attempts – decision provided by security logic

HVMI’s key challenges

- **bridge the semantic gap** – correlate raw 4K physical memory pages with meaningful OS data structures and operations
  - what ***objects*** are inside a guest VM?
  - what ***operations*** are being performed inside a guest VM?
- ensure acceptable / **low performance overhead**
  - forward lots of mem-event notifications with low overhead to engine
  - intercept only meaningful events
  - handle events quickly (analysis, re-execution / emulation, …)

User mode memory introspection

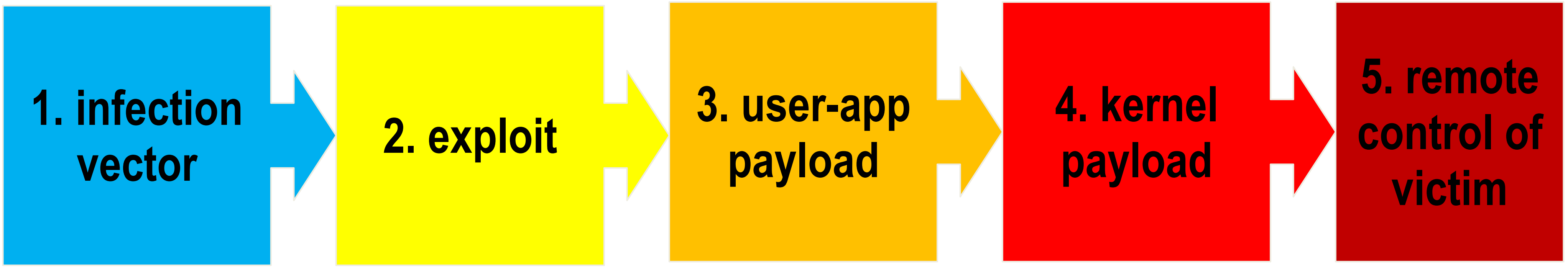
- monitor user applications (such as web-browsers, Microsoft* Office, Adobe* Reader, …) for
  - detection of code injection
  - detection of function detouring
  - enforcement of generic Write-XOR-eXecute (W⊕X) policy
- injection of remediation tools into the guest runtime on-the-fly (no help from ‘within’ guest needed)

How can UM HVMI improve security?

Spear phishing
Drive-by downloads
Trojans

Code injection → Zeus, Epic
Turla, Energetic Bear, …
API hooking → Dyreza,
GameOver, ….

Espionage &
data exfiltration
Identity theft
Sabotage



1. infection vector

2. exploit

3. user-app payload

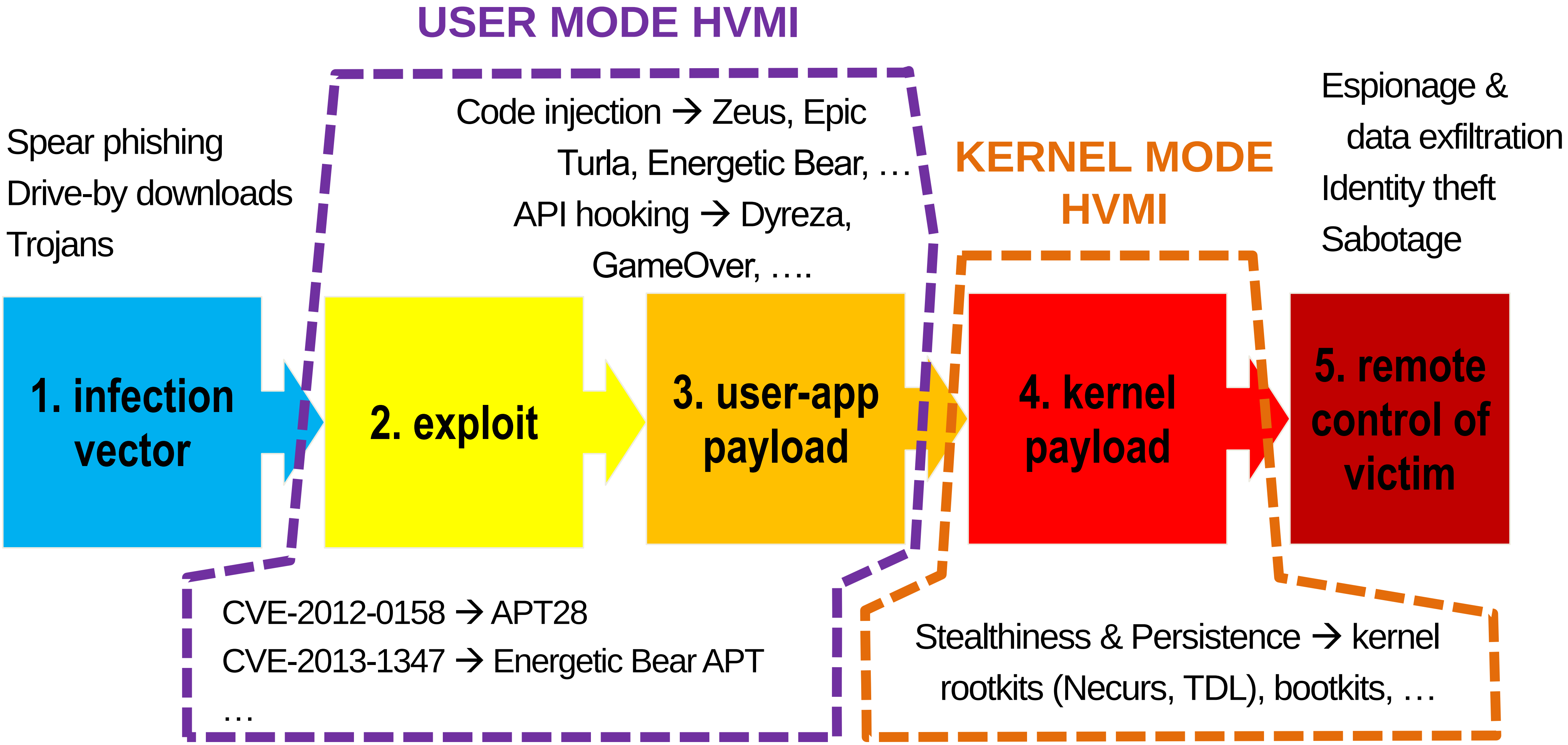
4. kernel payload

5. remote control of victim

CVE-2012-0158 → APT28
CVE-2013-1347 → Energetic Bear APT
…

Stealthiness & Persistence → kernel
rootkits (Necurs, TDL), bootkits, …

How can UM HVMI improve security?



USER MODE HVMI

Spear phishing
Drive-by downloads
Trojans

Code injection → Zeus, Epic
Turla, Energetic Bear, …
API hooking → Dyreza,
GameOver, ….

KERNEL MODE
HVMI

Espionage &
data exfiltration
Identity theft
Sabotage

1. infection vector

2. exploit

3. user-app payload

4. kernel payload

5. remote control of victim

CVE-2012-0158 → APT28
CVE-2013-1347 → Energetic Bear APT
…

Stealthiness & Persistence → kernel
rootkits (Necurs, TDL), bootkits, …

UM HVMI is STRONGLY ISOLATED (enforced by hardware) and provides GENERIC detection mechanisms

Dedicated VM vs Live VM introspection

	Dedicated VM (asynchronous image, on premise, in-lab, …)	Live VM Introspection	Mitigation approaches
Mem-event delivery time	not an issue	significant impact	Intel* Broadwell • ~400 ticks solely for the CPU round-trip • #VE avoid VMexits

Dedicated VM vs Live VM introspection

	Dedicated VM (asynchronous image, on premise, in-lab, …)	**Live VM Introspection**	**Mitigation approaches**
Mem-event delivery time	not an issue	significant impact	Intel* Broadwell • ~400 ticks solely for the CPU round-trip • #VE avoid VMexits
Overhead due to coarse grained 4K memory interception / filtering (unwanted mem-events)	usually not an issue	**very significant impact**	• today N/A • could be solved by future CPUs???

Dedicated VM vs Live VM introspection

	Dedicated VM (asynchronous image, on premise, in-lab, …)	**Live VM Introspection**	**Mitigation approaches**
Mem-event delivery time	not an issue	significant impact	Intel* Broadwell • ~400 ticks solely for the CPU round-trip • #VE avoid VMexits
Overhead due to coarse grained 4K memory interception / filtering (unwanted mem-events)	usually not an issue	**very significant impact**	• today N/A • could be solved by future CPUs???
Event processing time (decoding, security decision logic, emulation…)	can afford lengthy processing	very limited time	good logic, caching

Dedicated VM vs Live VM introspection

	Dedicated VM (asynchronous image, on premise, in-lab, …)	**Live VM Introspection**	**Mitigation approaches**
Mem-event delivery time	not an issue	significant impact	Intel* Broadwell • ~400 ticks solely for the CPU round-trip • #VE avoid VMexits
Overhead due to coarse grained 4K memory interception / filtering (unwanted mem-events)	usually not an issue	**very significant impact**	• today N/A • could be solved by future CPUs???
Event processing time (decoding, security decision logic, emulation …)	can afford lengthy processing	very limited time	good logic, caching
Availability of 3rd party analysis tools, external info and scripting	yes, many of them (PDB metadata, scripting, Volatility, …)	no, can't afford time overhead	N/A

Kernel mode vs User mode introspection

	Kernel Mode introspection	User Mode introspection	Mitigation approaches
Overcoming the semantic gap	challenging	more challenging (shared memory, multiple VA spaces, …)	N/A

Kernel mode vs User mode introspection

	Kernel Mode introspection	**User Mode introspection**	**Mitigation approaches**
Overcoming the semantic gap	challenging	more challenging (shared memory, multiple VA spaces, …)	N/A
Page protection lifetime	mostly static	**highly dynamic (follows process lifetime)**	detailed page table monitoring

Kernel mode vs User mode introspection

	Kernel Mode introspection	**User Mode introspection**	**Mitigation approaches**
Overcoming the semantic gap	challenging	more challenging (shared memory, multiple VA spaces, …)	N/A
Page protection lifetime	mostly static	**highly dynamic (follows process lifetime)**	detailed page table monitoring
Accessing swapped out pages	rarely an issue	significant / constant issue	#PF injection

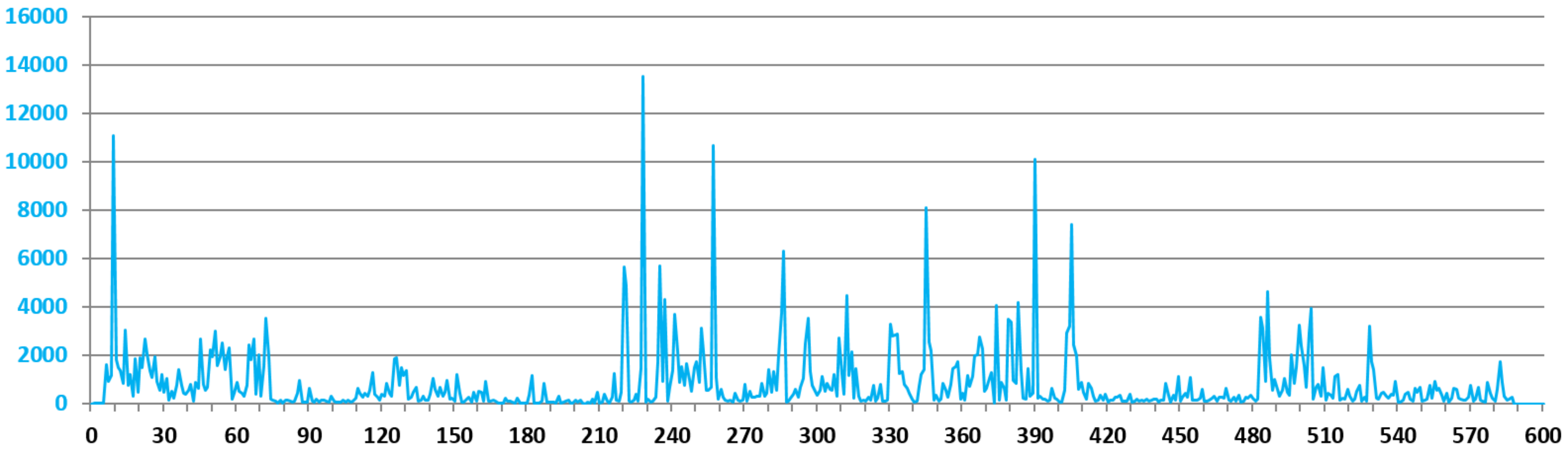
Kernel mode vs User mode introspection

	Kernel Mode introspection	**User Mode introspection**	**Mitigation approaches**
Overcoming the semantic gap	challenging	more challenging (shared memory, multiple VA spaces, …)	N/A
Page protection lifetime	mostly static	**highly dynamic (follows process lifetime)**	detailed page table monitoring
Accessing swapped out pages	rarely an issue	significant / constant issue	#PF injection
CPU page walker A/D bit updates impact on guest page monitoring	not an issue / small impact	**significant issue for memory intensive workloads**	• today N/A • could be solved by future CPUs ???

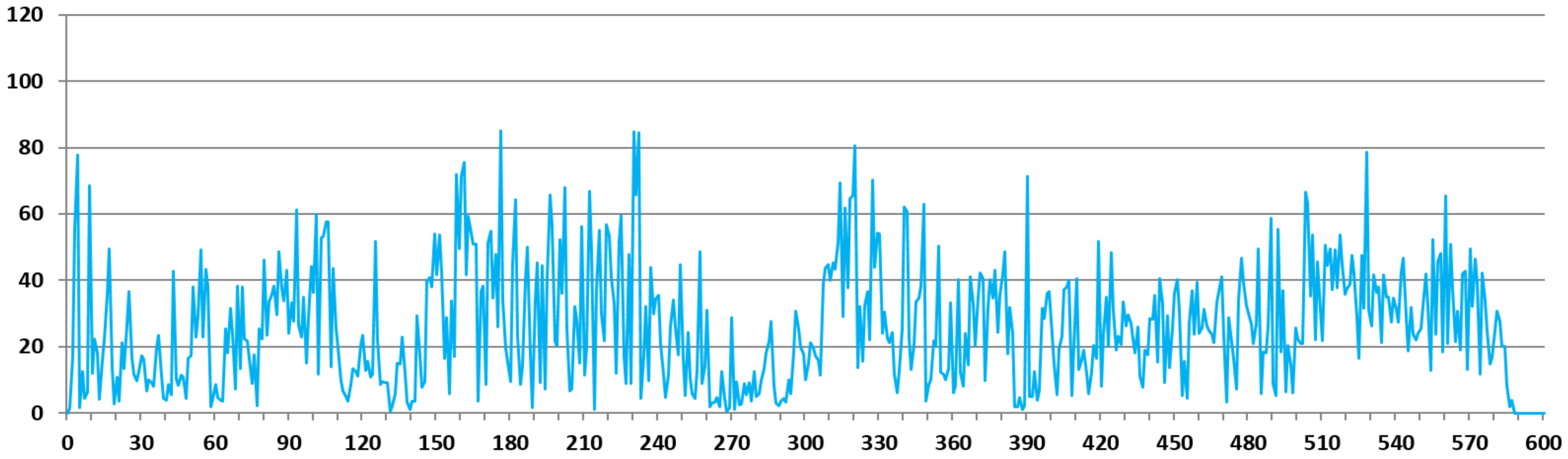
Limiting factors

VMexits due to CPU page walker A/D bit update

VMexits due to EPT violation induced by CPU page-walker updates of guest A/D bits



Percentage of A/D bit update generated EPT violations out of all violations



source: Bitdefender analysis

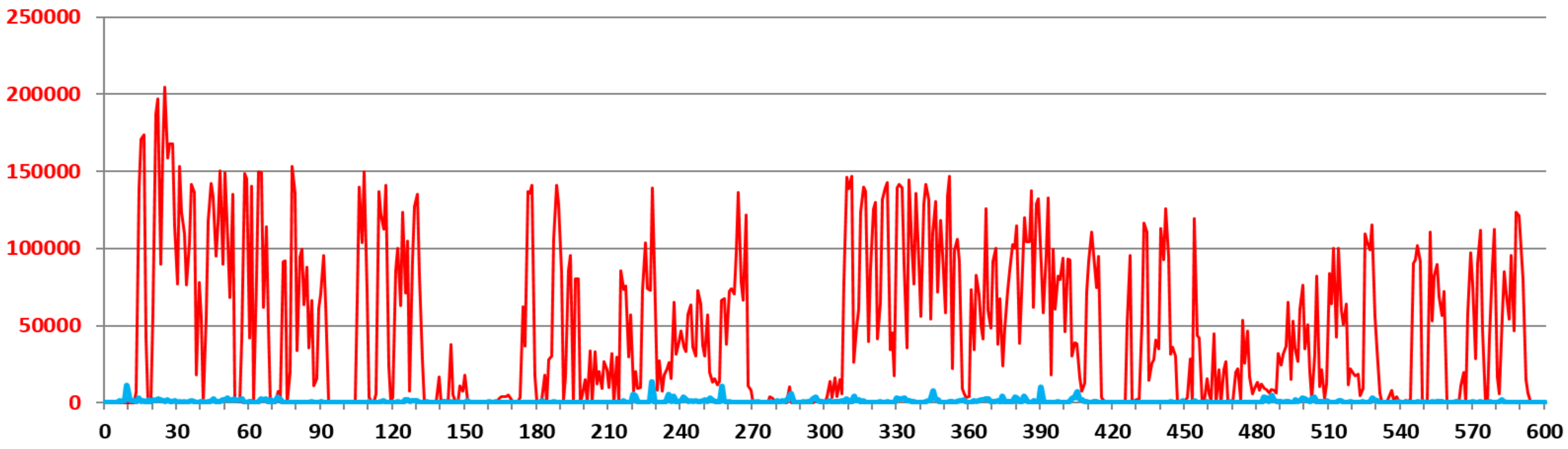
Typical office applications workload

(e.g. web browsing, document editing, …)

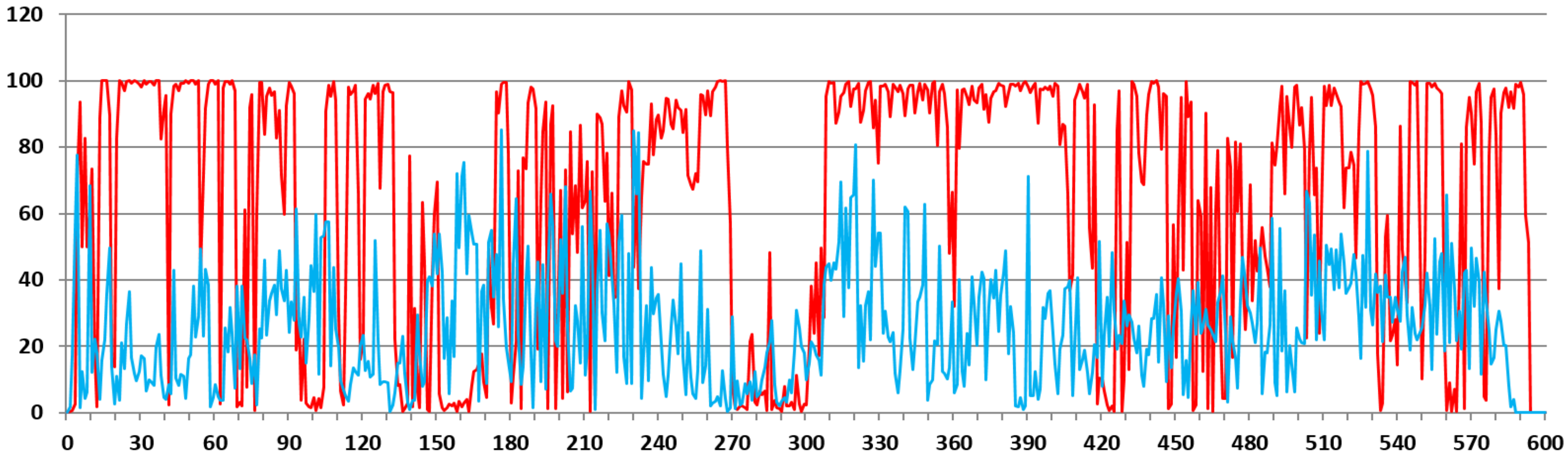
Limiting factors

VMexits due to CPU page walker A/D bit update

VMexits due to EPT violation induced by CPU page-walker updates of guest A/D bits



Percentage of A/D bit update generated EPT violations out of all violations



source: Bitdefender analysis

Typical office applications workload

(e.g. web browsing, document editing, …)

Heavy memory workload

(e.g. intensive allocations, many process starts, …)

Limiting factors

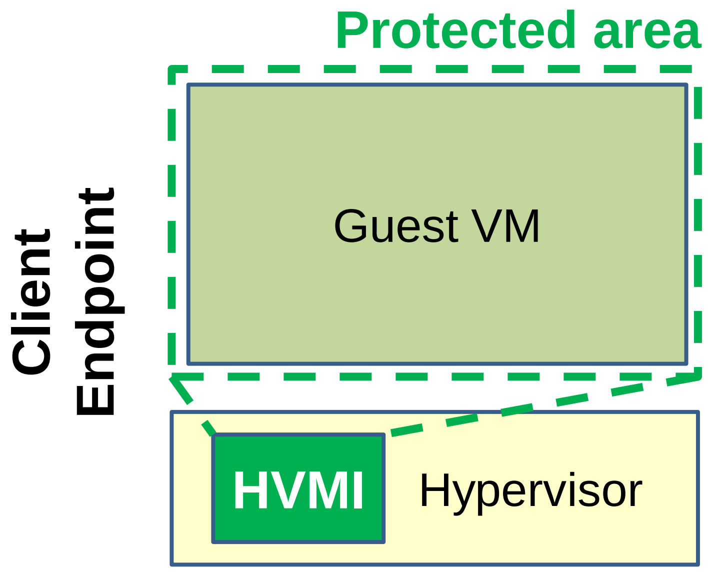
Instruction decoding – VMexit frequency

Instruction	Average %	Win 8.1 x64	Win 8 x86	Win 7 x86	Win 7 SP1 x64
MOV	**94.42**	746583	902462	401610	705405
CMPXCHG	**1.57**	42499	1558	3631	156
XADD	**0.98**	92	6250	14320	378
BTR	**0.56**	431	1640	8978	219
XOR	**0.34**	5590	2	118	4523
CMPXCH8B	**0.26**	51	878	2574	2597
INC	**0.15**	135	718	2027	373
BTS	**0.11**	1051	11	1273	41
DEC	**0.09**	433	1648	515	2
MOVZX	**0.06**	575	36	18	1221
All Other	**1.47**	4185	13364	15320	3609

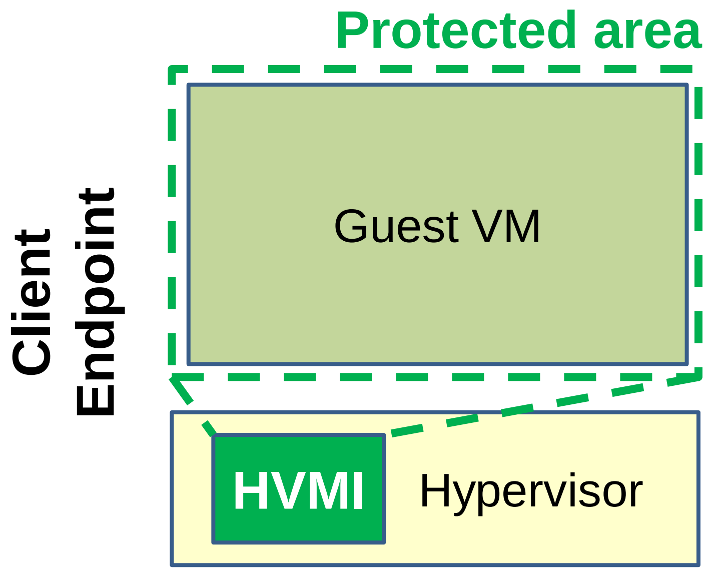
Total exits for each OS	801625	928567	450384	718524

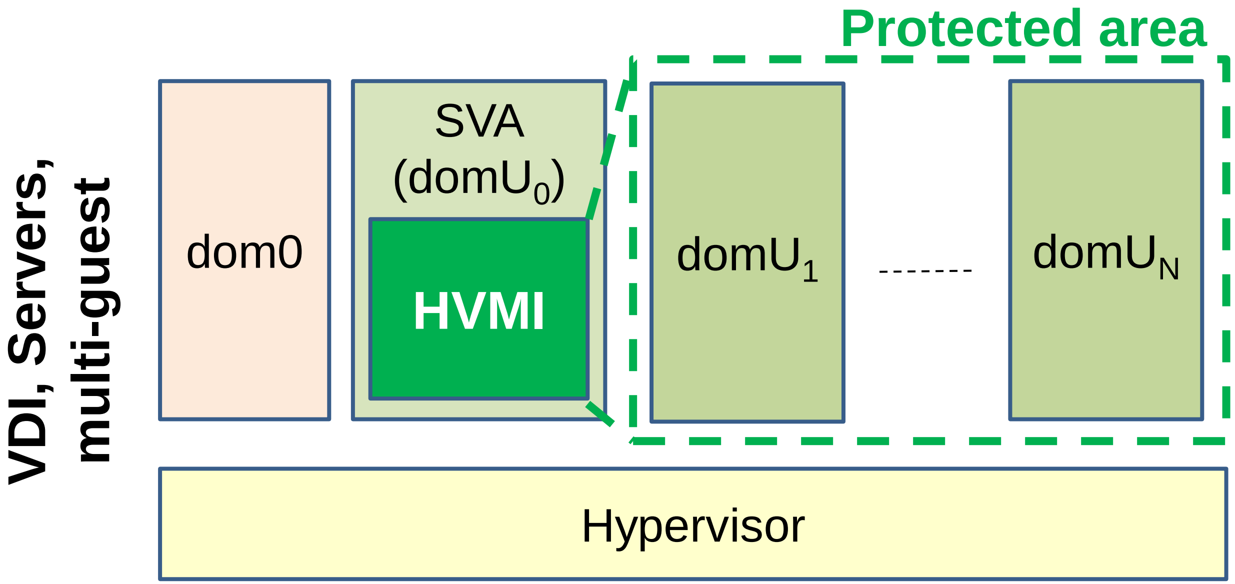
source: Bitdefender analysis

Introspection use-case scenarios

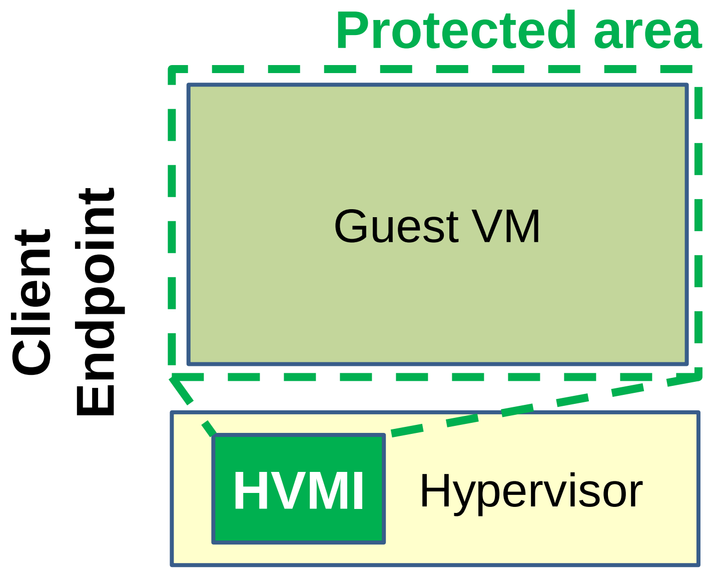


Introspection use-case scenarios





Introspection use-case scenarios



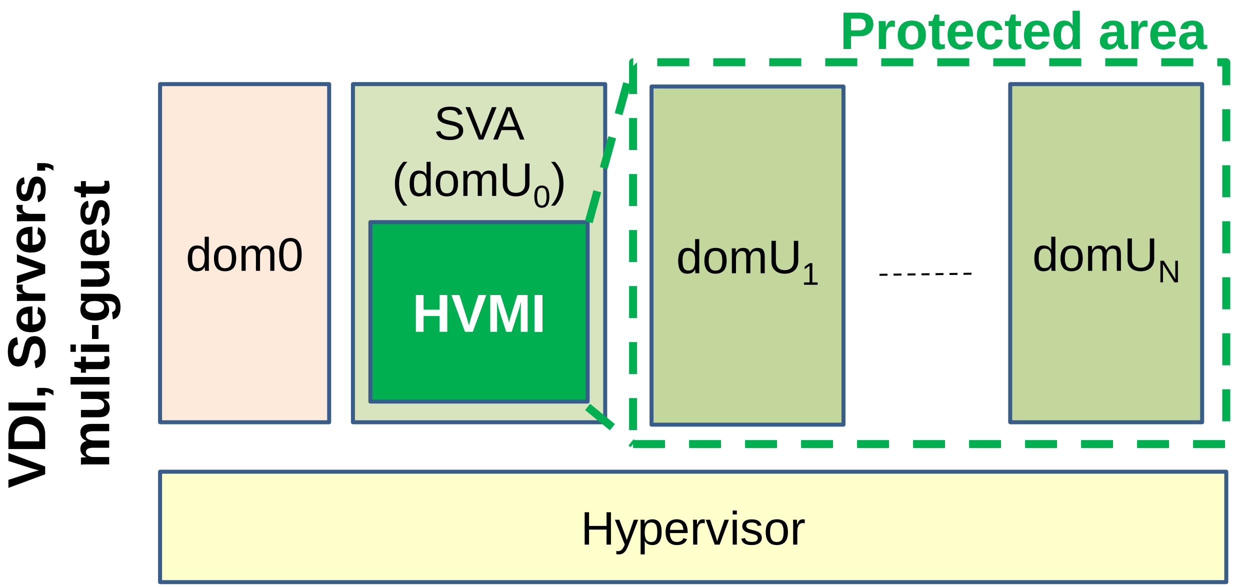
Client Endpoint

Protected area

Guest VM

HVMI

Hypervisor



VDI, Servers, multi-guest

Protected area

dom0

SVA (domU$_0$)

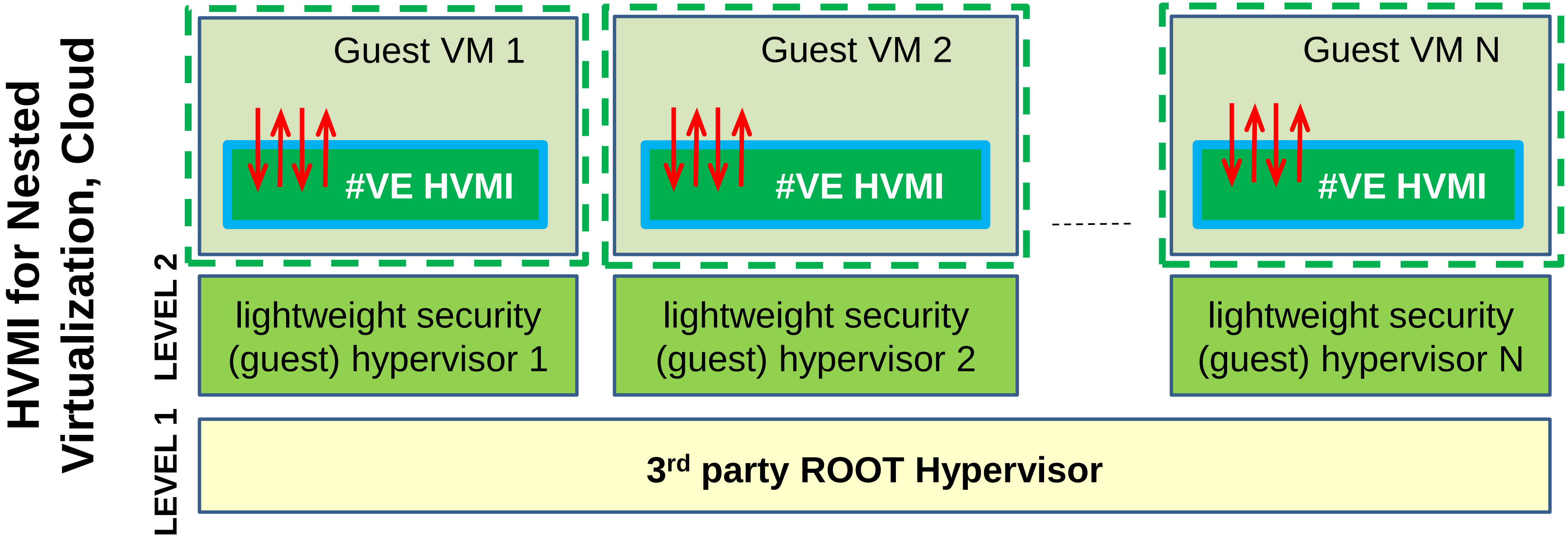
HVMI

domU$_1$

----

domU$_N$

Hypervisor



HVMI for Nested Virtualization, Cloud

Guest VM 1

#VE HVMI

Guest VM 2

#VE HVMI

Guest VM N

#VE HVMI

LEVEL 2

lightweight security (guest) hypervisor 1

lightweight security (guest) hypervisor 2

lightweight security (guest) hypervisor N

LEVEL 1

3rd party ROOT Hypervisor

Final thoughts

- **HVMI can be deployed today on a wide range of platforms**
  - cloud VMs, servers, VDI, endpoint clients (PCs, laptops, tablets)
  - Windows / Linux, 32 / 64 bit, x86 / ARM
  - kernel / user mode
  - in-hypervisor, Intel* #VE based, nested deployments
- **user mode introspection is very effective against a wide number of attacks, providing generic and strongly isolated security**
- user mode HVMI is good for typical office workloads, but there is room for improvement for heavy memory workload scenarios
  - this is an open research area, ideas are welcome ☺

Q&A

Thank you!

VMworld 2015 USA, August 30 – September 3, San Francisco

- live demos with Bitdefender HVMI on VMware* vSphere

Intel Developer Forum 2015 USA, August 18-20 San Francisco

- technical session talk on HVMI
- live demos with Bitdefender HVMI on Citrix* XenServer

Bitdefender®

* Names and brands might be claimed as the property of their respective owners.