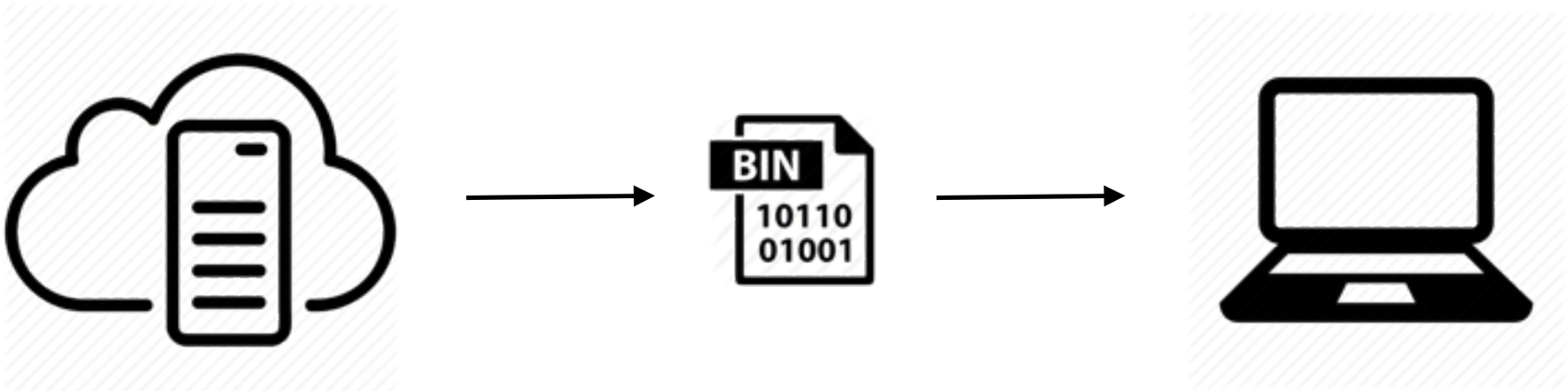
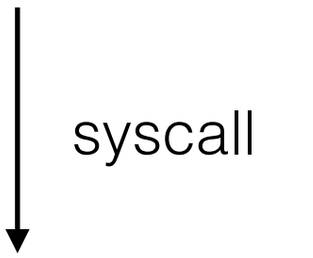
Hvexec: Safe Execution of Untrusted x86 Machine Code Using Hardware-Assisted Virtualization

Yi-Fan Zhang

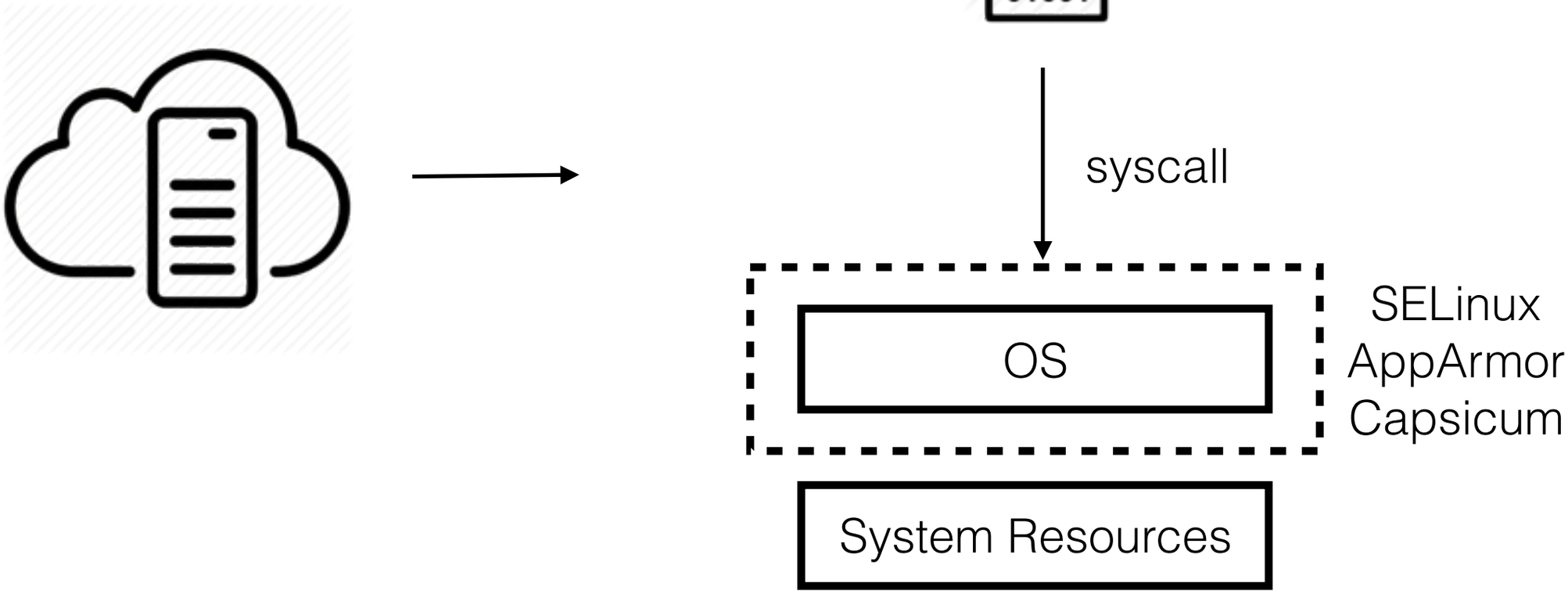

BIN
10110
01001

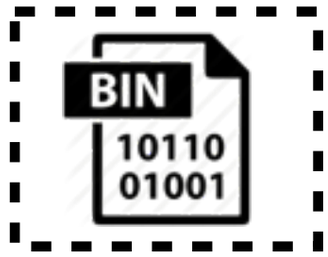

BIN
10110
01001


syscall


OS


System Resources


BIN
10110
01001
syscall
OS
SELinux
AppArmor
Capsicum
System Resources



Managed Runtimes
Binary Translation

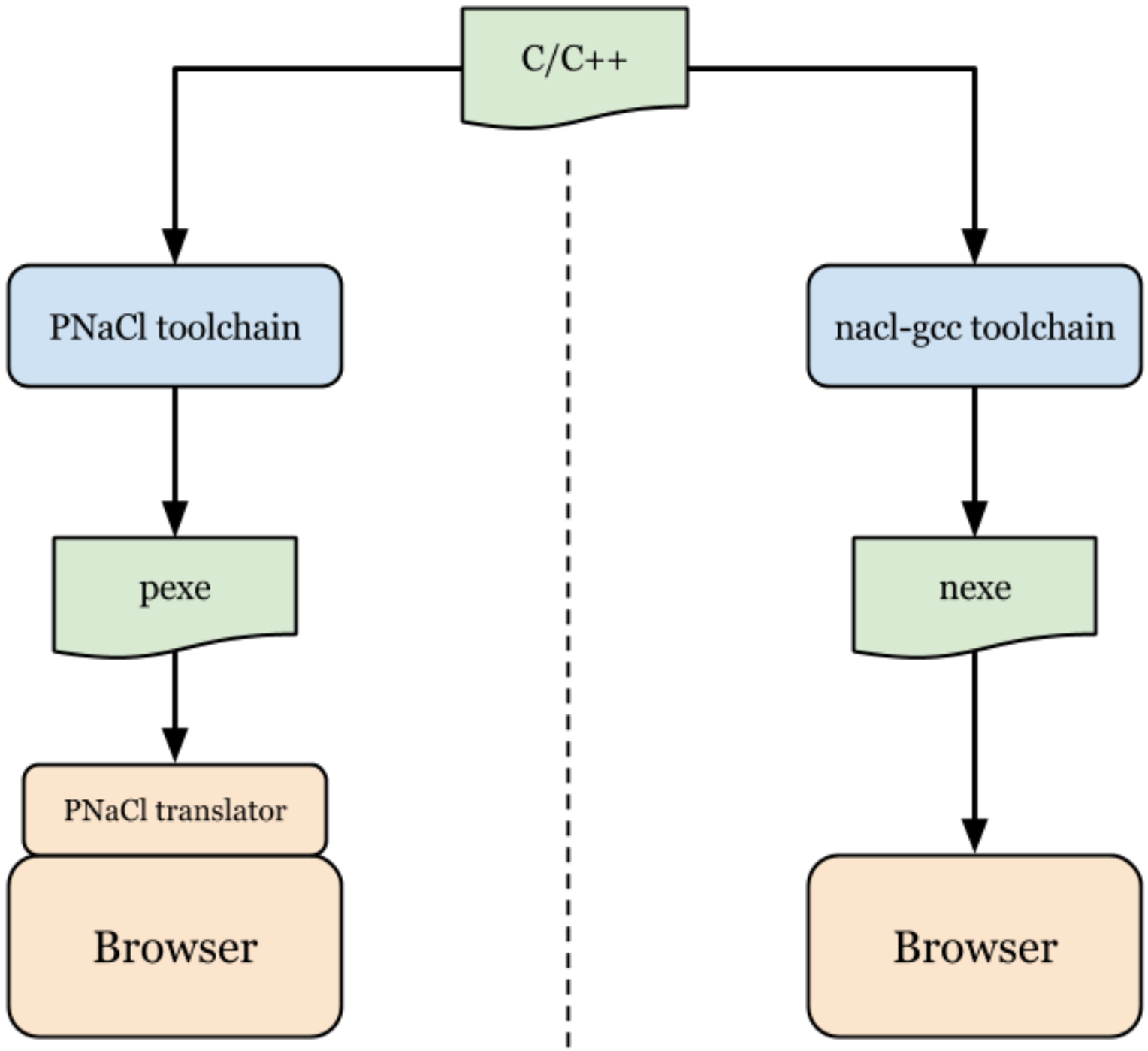
syscall

OS

System Resources

NaCl

write native code for the web



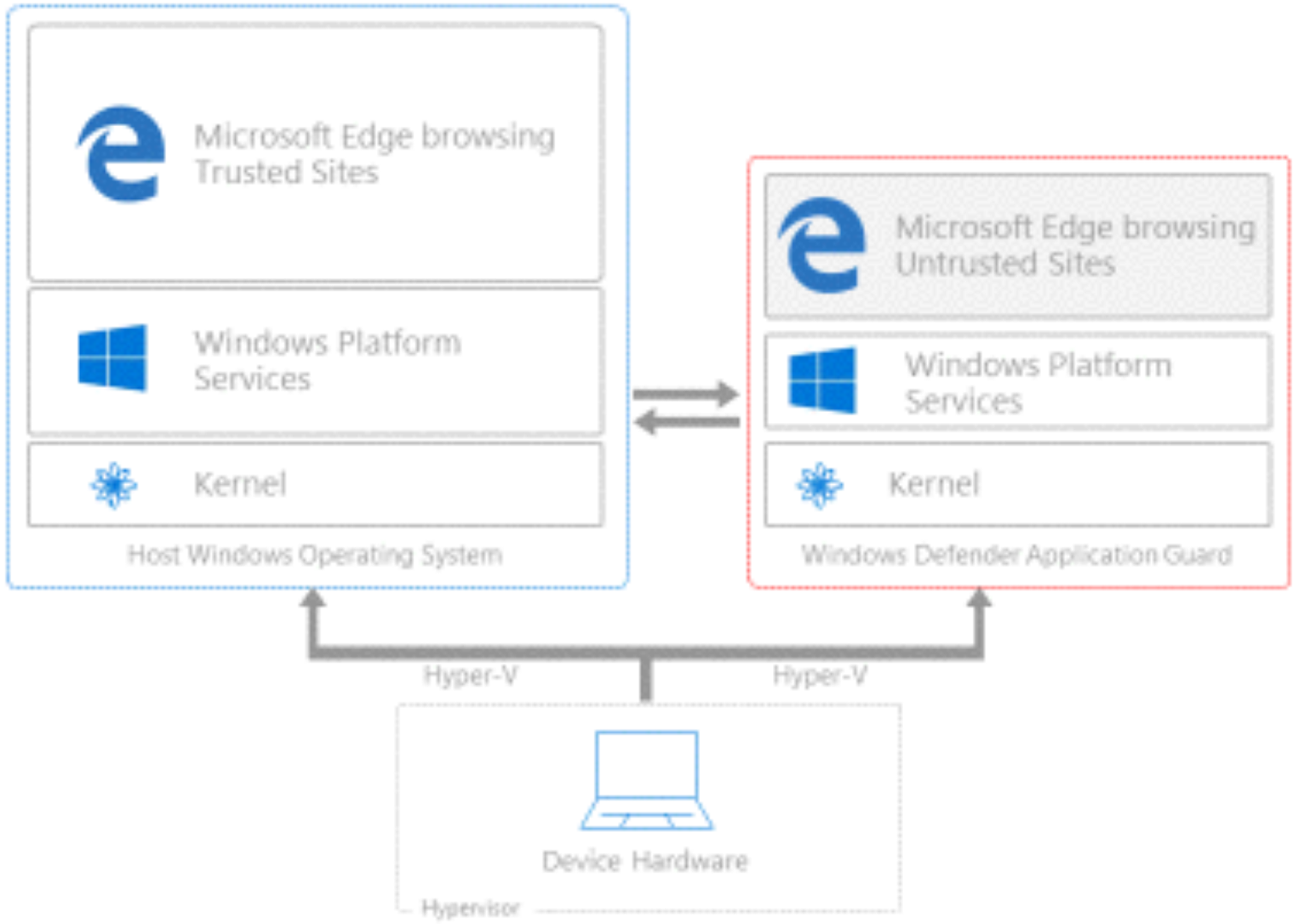
Workflow for web applications

Workflow for the web-store and plug-ins

C1	Once loaded into the memory, the binary is not writable, enforced by OS-level protection mechanisms during execution.
C2	The binary is statically linked at a start address of zero, with the first byte of text at 64K.
C3	All indirect control transfers use a `nacljmp` pseudo-instruction (defined below).
C4	The binary is padded up to the nearest page with at least one `hlt` instruction (0xf4).
C5	The binary contains no instructions or pseudo-instructions overlapping a 32-byte boundary.
C6	All *valid* instruction addresses are reachable by a fall-through disassembly that starts at the load (base) address.
C7	All direct control transfers target valid instructions.

Table 1: Constraints for NaCl binaries.

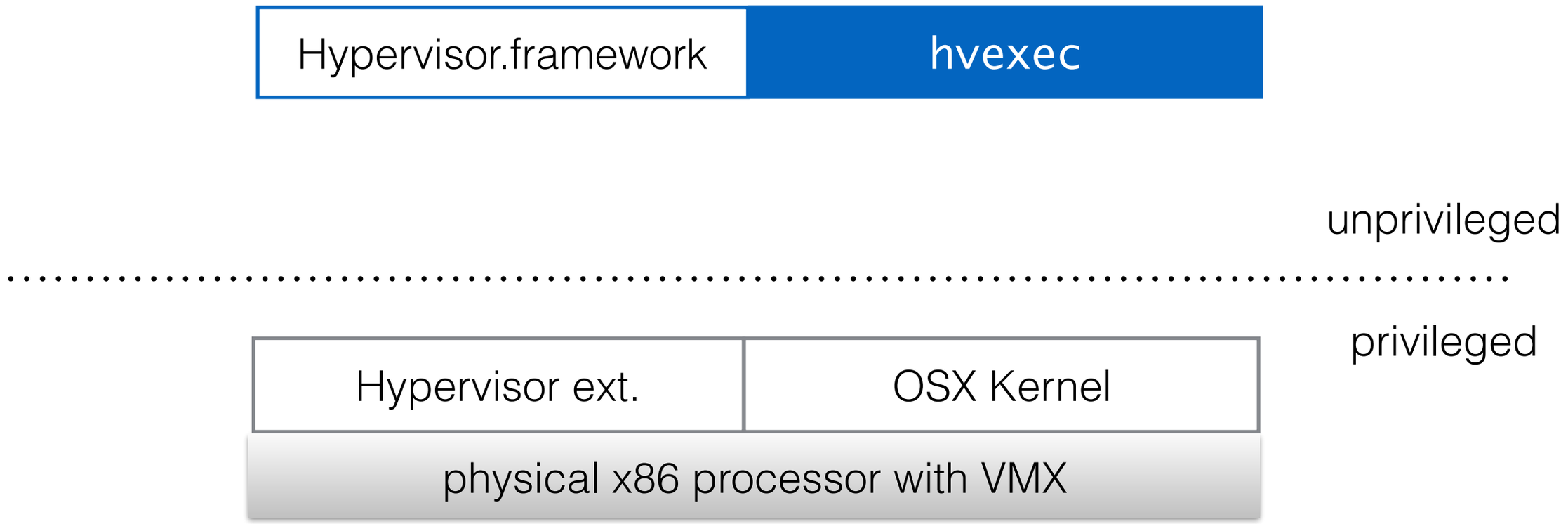
Microsoft Application Guard

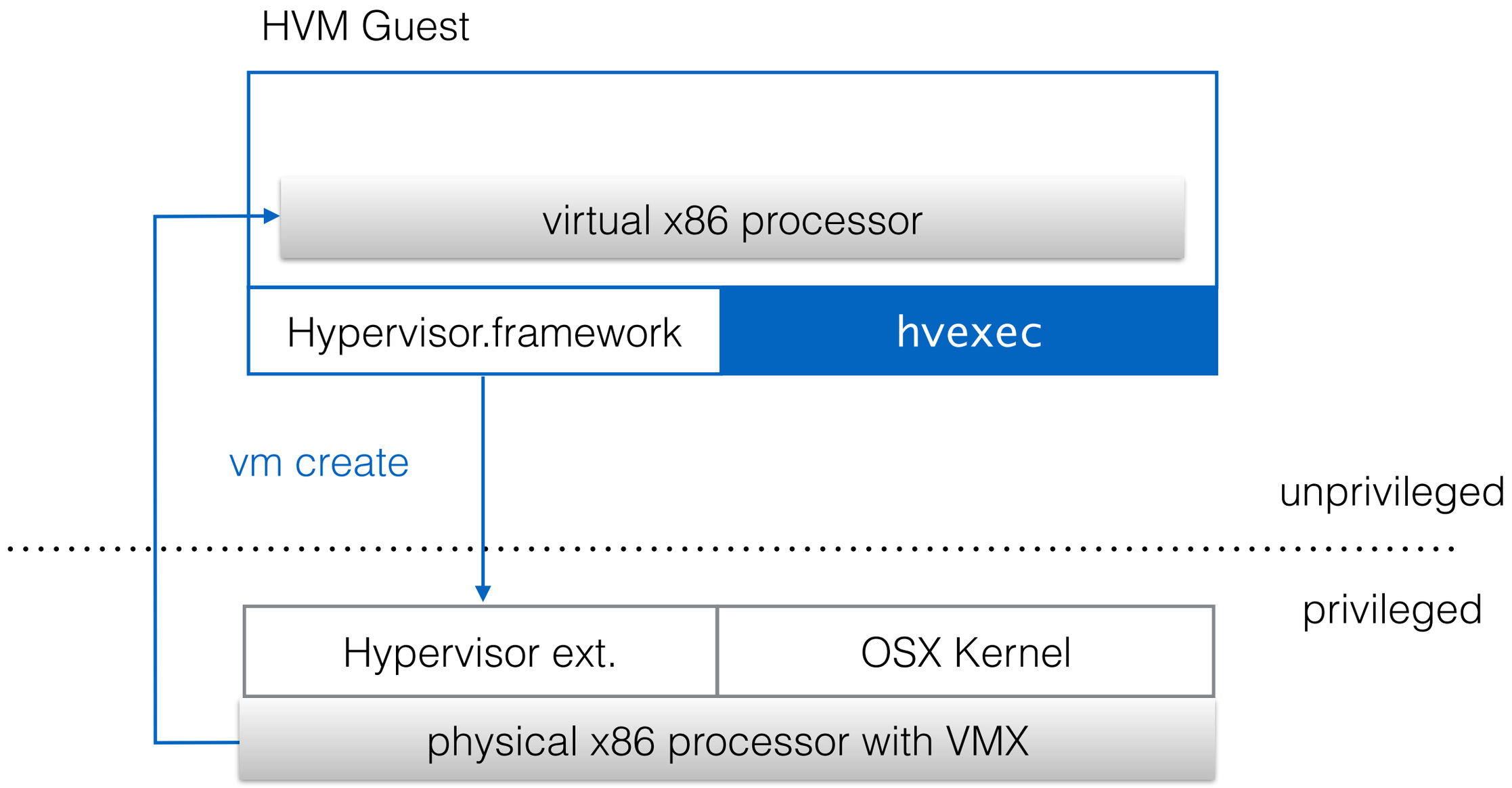

Microsoft Edge browsing
Trusted Sites
Windows Platform
Services
Kernel
Host Windows Operating System
Microsoft Edge browsing
Untrusted Sites
Windows Platform
Services
Kernel
Windows Defender Application Guard
Hyper-V
Hyper-V
Device Hardware
Hypervisor

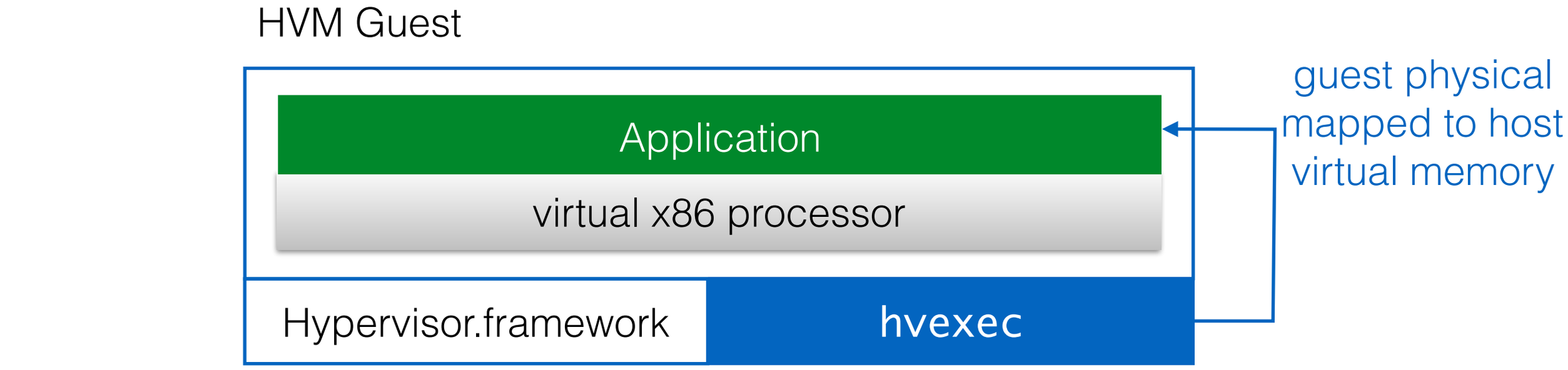
Hvexec

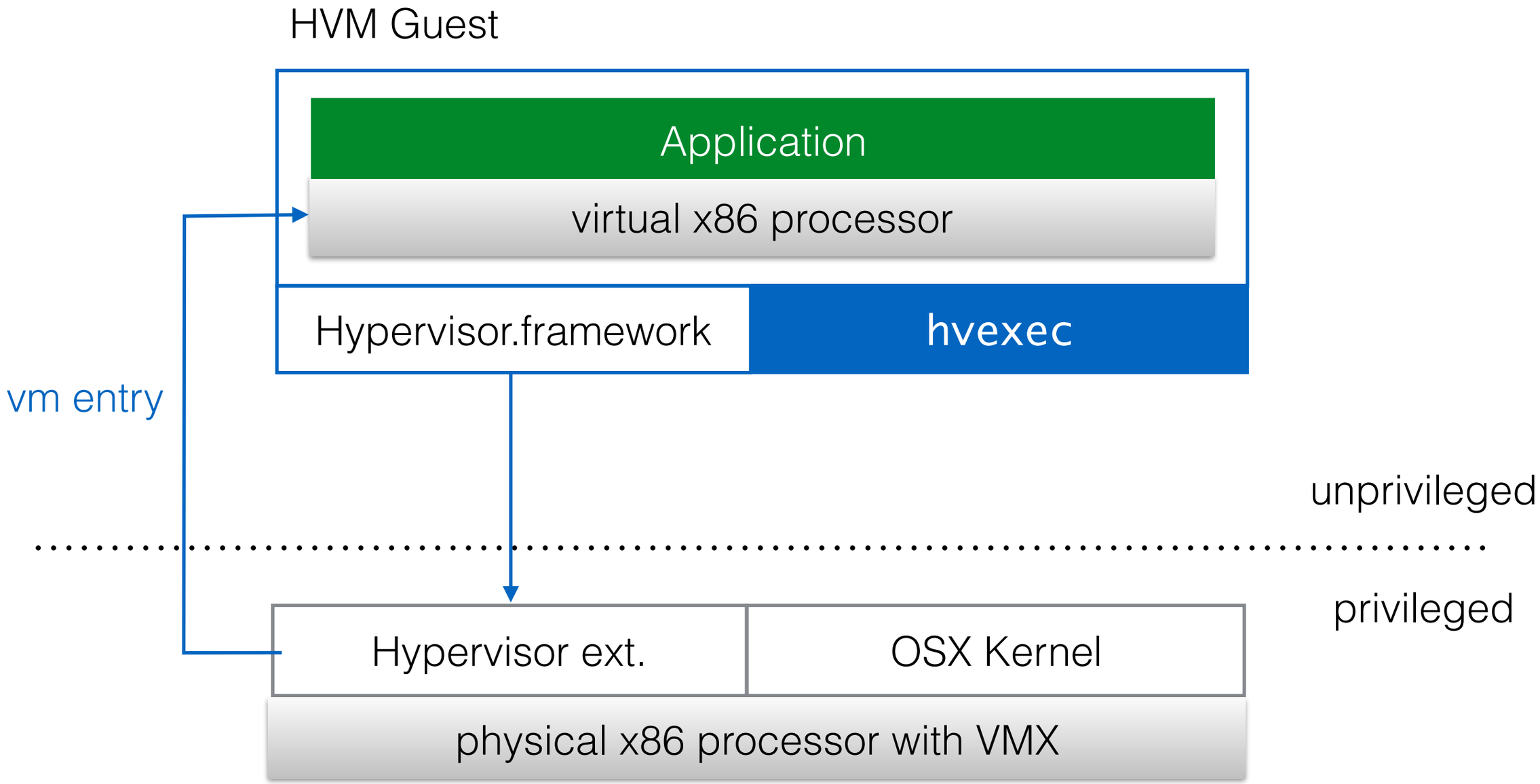
Hvexec ≈ NaCl

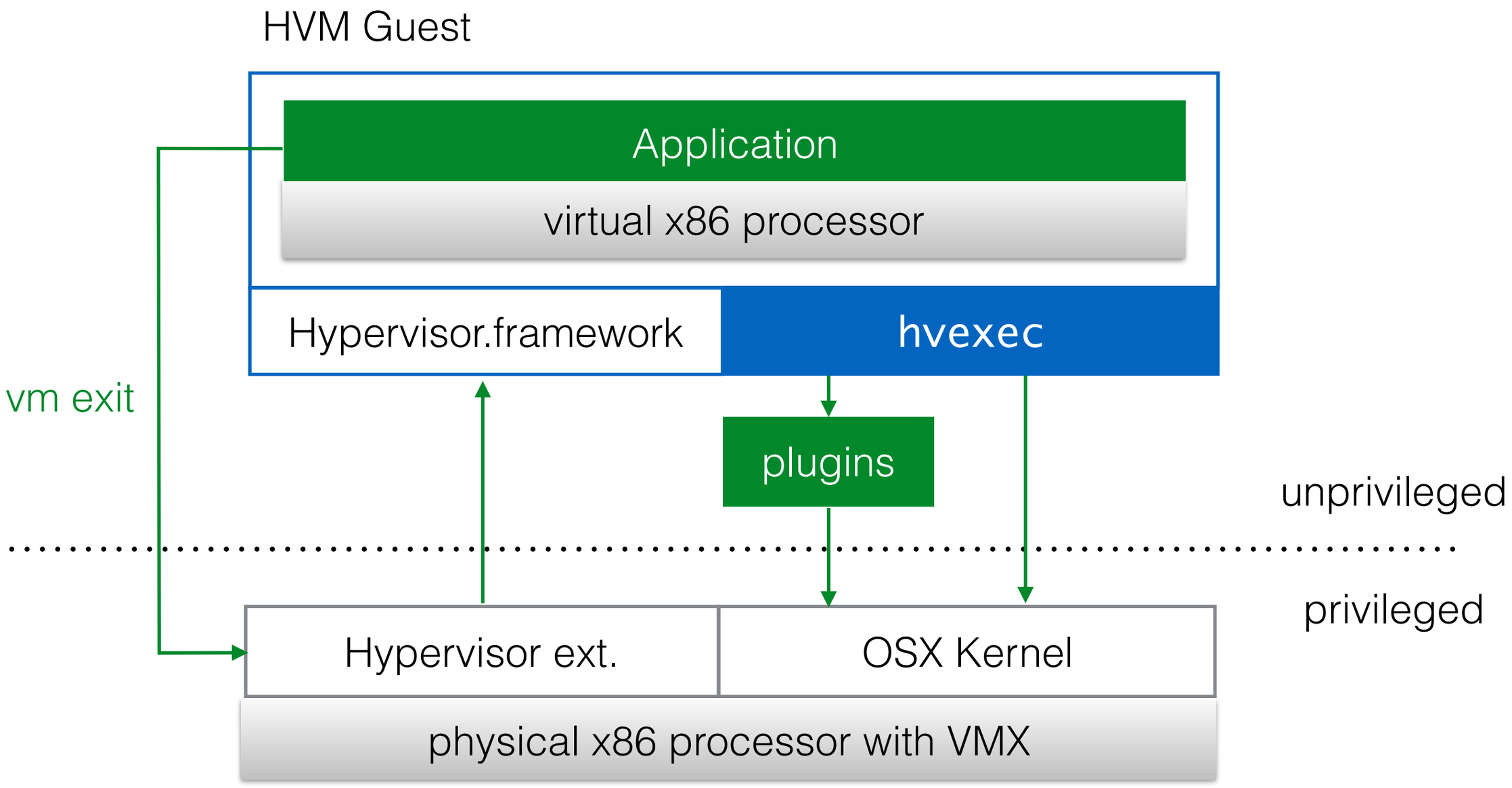
Hvexec ≈ NaCl + Windows App Guard


Hypervisor.framework
hvexec
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX


HVM Guest
virtual x86 processor
Hypervisor.framework
hvexec
vm create
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX


HVM Guest
Application
virtual x86 processor
Hypervisor.framework
hvexec
guest physical
mapped to host
virtual memory
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX


HVM Guest
Application
virtual x86 processor
Hypervisor.framework
hvexec
vm entry
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX


HVM Guest
Application
virtual x86 processor
Hypervisor.framework
hvexec
vm exit
plugins
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX

Demo

References

- Intel® 64 and IA-32 Architectures Software Developer's Manual. Volume 3 (3A, 3B, 3C & 3D): System Programming Guide

- Yee, Bennet, et al. "Native client: A sandbox for portable, untrusted x86 native code." 2009 30th IEEE Symposium on Security and Privacy. IEEE, 2009.

- Images

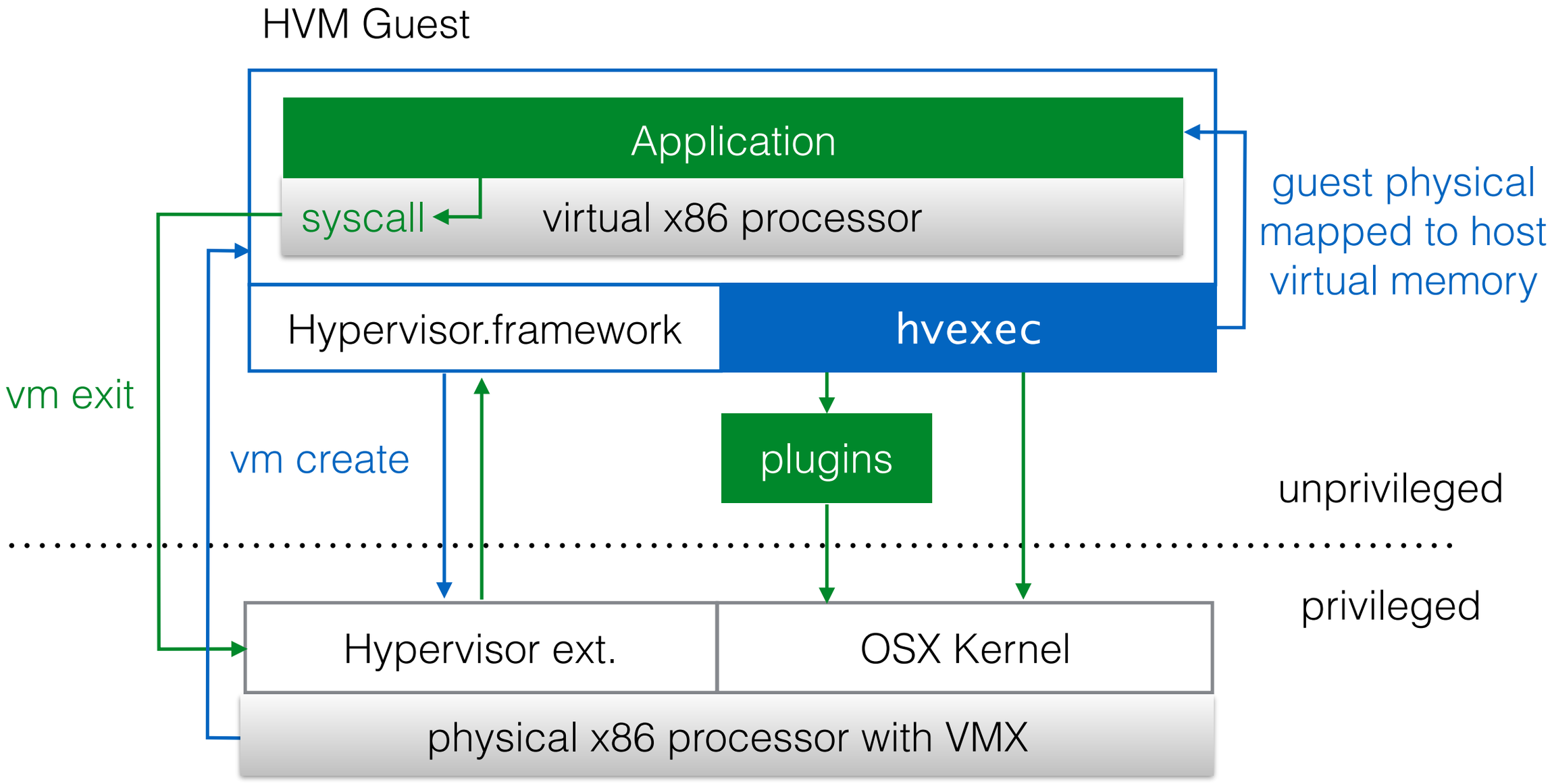
  - http://polimeraus.blogspot.com/2012/11/native-client-teknik-baks-1.html

  - https://xakep.ru/2014/10/08/google-native-client/

  - https://blogs.windows.com/msedgedev/2016/09/27/application-guard-microsoft-edge/#IgwQHlS8OCJAX8dy.97

  - http://www.freepik.com

  - https://www.iconfinder.com


HVM Guest
Application
syscall
virtual x86 processor
Hypervisor.framework
hvexec
guest physical
mapped to host
virtual memory
vm exit
vm create
plugins
unprivileged
privileged
Hypervisor ext.
OSX Kernel
physical x86 processor with VMX