Hyper-V Internals

Zafiyet Araştırmacısı Perspektifinden Hyper-V'ye İlk Bakış

Alperen Şirin **15.10.2022**

Gündem

- Hipervizör nedir?
- Hedef olarak neden hipervizörler ve neden Hyper-V?
- Hyper-V internals
- Saldırı senaryoları ve saldırı yüzeyi
- Hyper-V zafiyet araştırması tavsiyeleri

Hipervizör nedir?

- Sanal makine oluşturan, çalıştıran yazılım ve/veya donanımdır.
- Yönetim/bakım kolaylığı, güvenlik, enerji tasarrufu gibi avantajlar sağlamaktadır.
- Popüler işlemciler sanallaştırma uzantıları ile donanım desteği sağlamaktadır(Intel VT, AMD-V gibi).

Hipervizör nedir?

Hipervizör çeşitleri

- Type-1(baremetal) hipervizörler
  - Donanımla aracısız konuşurlar.
- Type-2 hipervizörler
  - Konak(host) işletim sistemi altında çalışırlar.
- Modern hipervizörlerde bu konseptler birbirine karışmış durumdadır.

Hedef olarak neden hipervizörler?

- Bulut altyapısının(büyük oranda) hipervizörlere dayanması
- Kurumsal BT altyapılarının önemli bir bölümünün sanal olması
- Sanallaştırmanın öneminin artış trendinde olması
- İlgi çekici konsept(zevkler, renkler…)
- İşletim sistemi mimarilerine de konsept olarak girmeye başlaması(Windows)

Hedef olarak neden Hyper-V?

- İyi tasarlanmış ve sıkı bir hedef olması(en az yetki prensibi)
- İlgi çekici mimari/konseptler(zevkler, renkler…)
- Windows mimarisinin parçası olması(Windows Virtual Secure Mode)
- Microsoft Hyper-V Bounty Program

Hyper-V internals

Hyper-V'ye özel terimler

- **Partition**: İçinde işletim sistemi çalışan ve izolasyonu hipervizör tarafından sağlanan mantıksal ünite.
- **"Enlightened" konuk**: "Para-virtualization" konseptinin Hyper-V karşılığı. Daha iyi sanallaştırma performansı için modifiye edilmiş ve Hyper-V farkındalığı olan konuk.
- **VID(Virtualization Infrastructure Driver)**: Partition, sanal işlemci ve hafıza yönetiminden sorumlu bileşen.

Hyper-V internals

Hyper-V'ye özel terimler

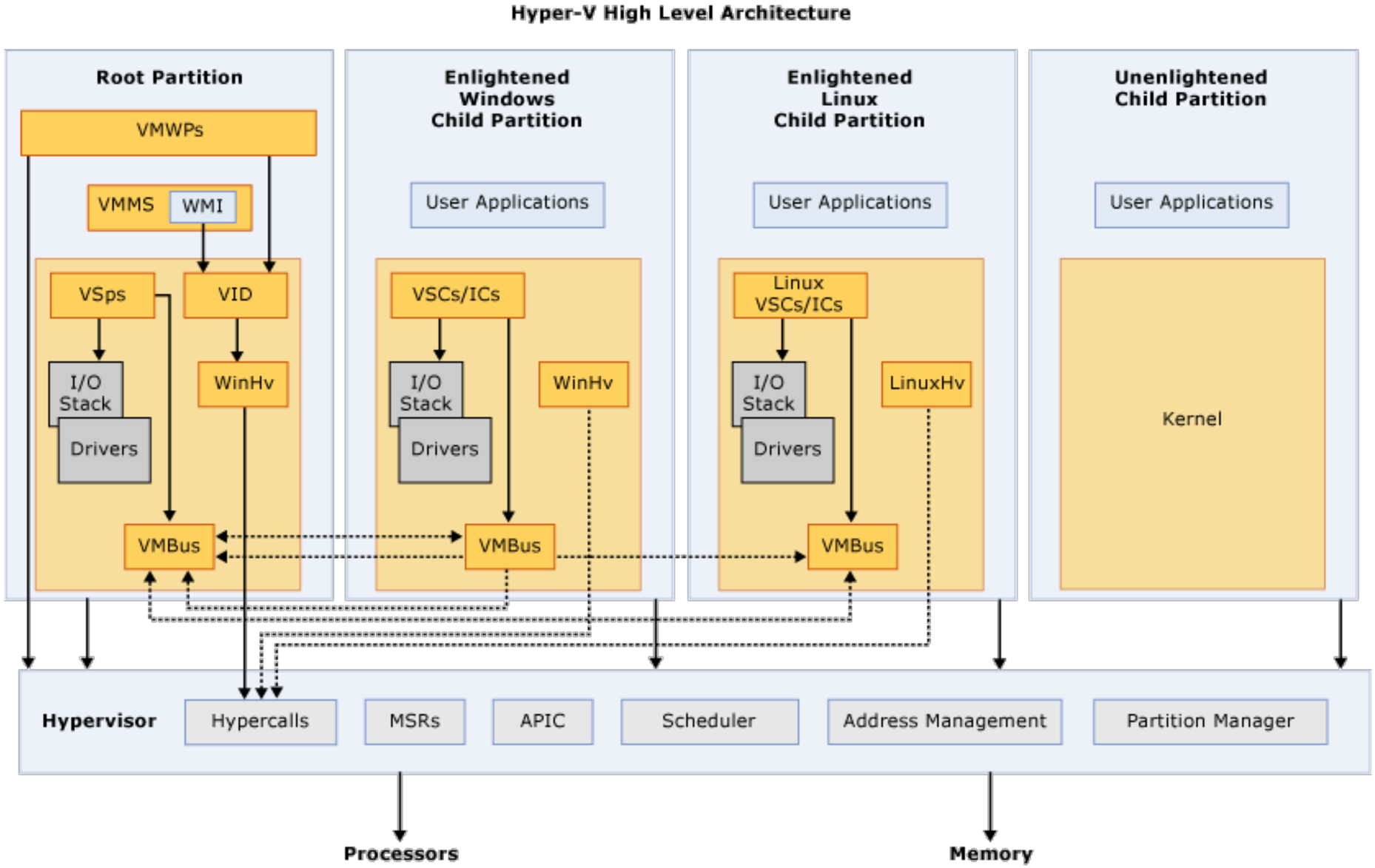
- **VMBus**: Çeşitli amaçlar için kullanılan partitionlar arası iletişim kanalı.
- **VSP(Virtualization Service Provider)**: Sentetik cihazları servis olarak sunan bileşen, kök partition kernel'da yaşar.
- **VSC(Virtualization Service Client)**: VSP ile VMBus kanalından konuşan VSP istemcileri, konuk partition kernel'da yaşar.
- **Hypercall**: Hipervizör ile iletişim kanalı.

Hyper-V internals

Hyper-V'ye özel terimler

- **VMWP(VM Worker Process)**: Kök partition temel userland bileşeni. Konuk partitionların yönetimi ile ilgili userland'de çalışan çoğu koda ev sahipliği eden process'tir. Her konuk partition için bir adet VMWP oluşturulur.

Hyper-V internals



Kaynak: https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/reference/hyper-v-architecture

Hyper-V internals

Hipervizör

- Hyper-V’nin hipervizörü donanımla aracısız etkileşir(type 1).
- Partition’ların fiziksel hafızalarını yönetir ve izolasyonlarını sağlar.
- Bazı instruction’ları araya girerek emüle eder.
- Konuklara gönderilen donanım interrupt’larını yönetir.
- Kısaca partition izolasyonunu sağlayan donanım bekçisi yazılım katmanıdır.
- Konuk partitionlar ile direkt iletişimi çok sınırlıdır.

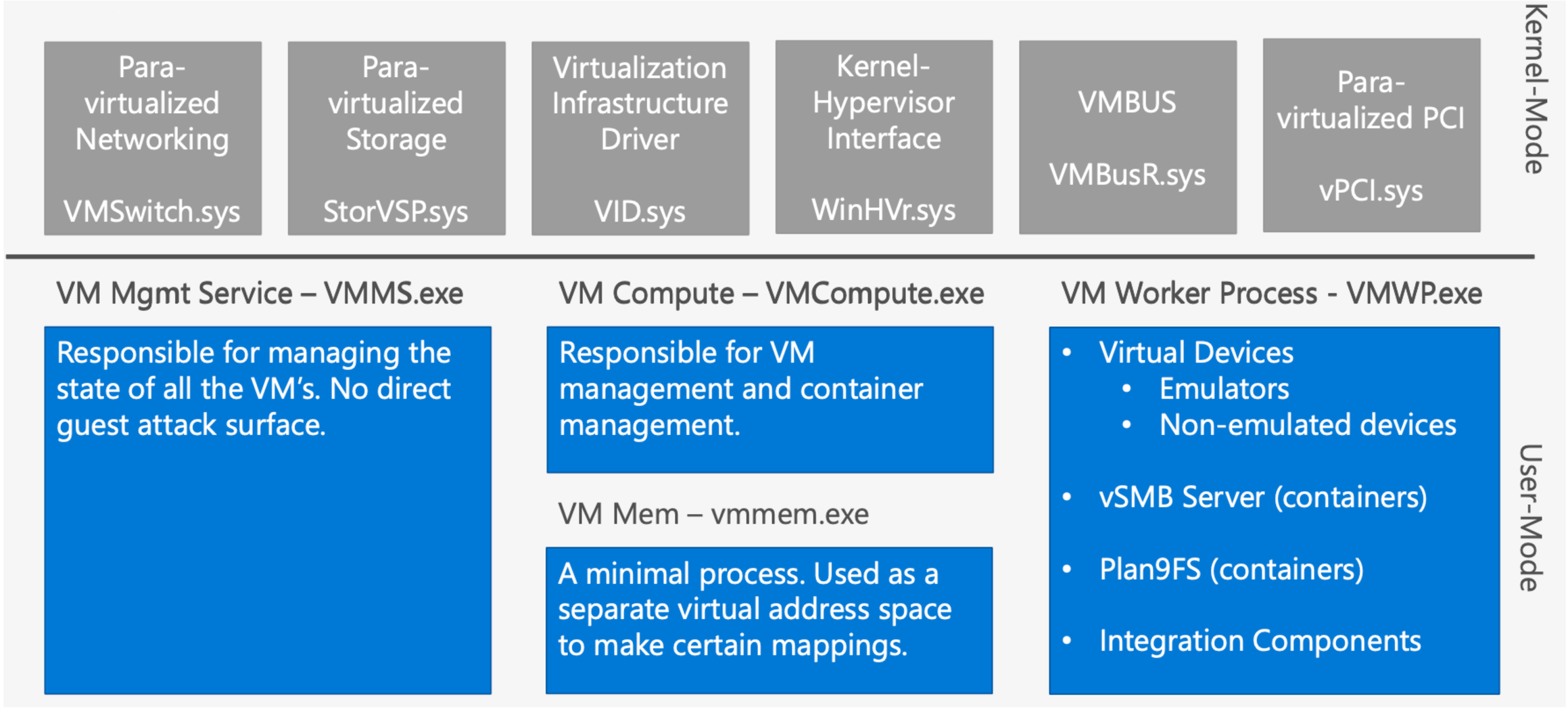
Hyper-V internals

Kök partition

- Özel bir partition’dır.
- Hipervizör ile konuk partitionlar arasındaki katmandır.
- Diğer partitionlar’ı oluşturur, yok eder.
- Sanal cihazları diğer partitionlar’ın kullanımına sunar.
- Fiziksel hafızaya ve fiziksel cihazlara erişimi olan tek partition’dır.
- Konuk perspektifinden saldırı yüzeyinin büyük bir kısmını barındırır.

Hyper-V internals

Kök partition(2018 Ağustos)



Kaynak: A Dive in to Hyper-V Architecture & Vulnerabilities, Nicolas Joly & Joe Bialek

Saldırı senaryoları

Konuk perspektifinden

- Konuktan konuğa.
- Konuktan kök partition userland bileşenlere.
- Konuktan kök partition kernel bileşenlere.
- Konuktan hipervizöre.

Saldırı senaryoları

Kök partition userland perspektifinden(zincirlemek için önemli)

- Kök partition userland’den kök partition kernel’a.
- Kök partition userland’den hipervizöre.

Saldırı senaryoları

Kök partition kernel perspektifinden(hyperhacking)

- Kök partition kernel bileşenlerden hipervizöre.

Saldırı yüzeyi

Konuktan konuğa

- Konuktan konuğa direkt saldırı yüzeyi yoktur.
- Kök partition üzerinden sınırlı saldırı yüzeyi vardır.

Saldırı yüzeyi

Konuktan kök partition userland bileşenlere

- Temel hedef saldırgan konuk için oluşturulmuş VMWP'tir(VMWP.exe).

IO Ports	• User-mode components can register for notifications when particular IO ports are written/read • Used to emulate hardware
MMIO	• Components can register GPA ranges as MMIO ranges, receive notifications when the ranges are written/read • Used to emulate hardware
VMBUS	• High-speed communication channel accessed through named pipes or sockets
Aperture	• Map guest physical addresses into the virtual address space of VMWP • Need to be careful to avoid shared-memory issues such as double-fetch
Read/Write Notifications	• Triggered when a specified GPA is read/written, EIP is not advanced (no emulation) • Used to track when pages are dirtied while live migrating (as an example)

Kaynak: A Dive in to Hyper-V Architecture & Vulnerabilities, Nicolas Joly & Joe Bialek

Saldırı yüzeyi

Konuktan kök partition kernel bileşenlere

- Para-virtualized cihazlar(özellikle VMSwitch.sys)
- VMBusR.sys
- VID.sys

Saldırı yüzeyi

Konuktan kök partition kernel bileşenlere

VMBUS	• High-speed communication channel accessed through via Kernel Mode Client Library (KMCL) abstraction layer
Extended Hypercalls	• Hypercalls that the hypervisor forwards directly to the VID • Very few
Aperture	• Host can map guest physical memory and interact with it • Rarely used by kernel

Intercept Handling	• Hypervisor forwards some intercepts it receives to the host for processing
	  • IO port read/write (does it need emulation?)
	  • EPT faults: is the memory paged out?, is that memory a virtual MMIO page?
	  • Etc.

Kaynak: A Dive in to Hyper-V Architecture & Vulnerabilities, Nicolas Joly & Joe Bialek

Saldırı yüzeyi

Konuktan hipervizöre

- Konuktan hipervizöre direkt saldırı yüzeyi sınırlıdır.

Hypercalls	• "System calls" of the hypervisor • Guest accessible hypercalls are documented as part of the Hyper-V TLFS • Some Hypercalls pass arguments via registers, others use physical pages (GPA in register)
Faults	• Triple fault, EPT page faults (i.e. permission faults, GPA not mapped, etc.) • This is how MMIO can be virtualized by VDEV's (fault on access to virtual MMIO range)
Instruction Emulation	• Attempt to execute instructions such as CPUID, RDTSC, RDPMC, INVLPG, IN, OUT, etc.
Register Access	• Attempt to read/write control registers, MSR's
Overlay Pages	• A way for the hypervisor to forcibly map a physical page in to a partition • Example: Hypercall code page • Primarily used to communicate data to a guest partition

Kaynak: A Dive in to Hyper-V Architecture & Vulnerabilities, Nicolas Joly & Joe Bialek

Hyper-V zafiyet araştırması tavsiyeleri

- TLFS(Hypervisor Top-Level Function Specification) referans kaynak
- Linux kaynak kodunda birçok konuk bileşenin kaynak kodu var(VMBus istemcisi, sentetik cihaz driverları ve daha fazlası). Birçok şeyi anlamayı, fuzzlamayı kolaylaştırıyor.
- Hyper-V ile ilgili birçok çalıştırılabilir dosya için semboller Microsoft tarafından yayınlanıyor(2018 Nisan'dan beri).
- Çok fazla açık kaynak, araştırma, makale var(5 sene öncesine kıyasla).
- İlham alınacak hazır araçlar mevcut(nyx, hAFL2, hyntrospect).

Kaynaklar

- https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/reference/hyper-v-architecture
- https://github.com/Microsoft/MSRC-Security-Research/blob/master/presentations/2018_08_BlackHatUSA/A%20Dive%20in%20to%20Hyper-V%20Architecture%20and%20Vulnerabilities.pdf
- TLFS v6.0b
- https://github.com/gerhart01/Hyper-V-Internals/blob/master/HyperResearchesHistory.md
- https://alisa.sh/slides/HypervisorVulnerabilityResearch2020.pdf