

2008年3月19日(水)
第2回セキュアVMシンポジウム

筑波大学 講師 品川高廣

セキュアVMの アーキテクチャ概要

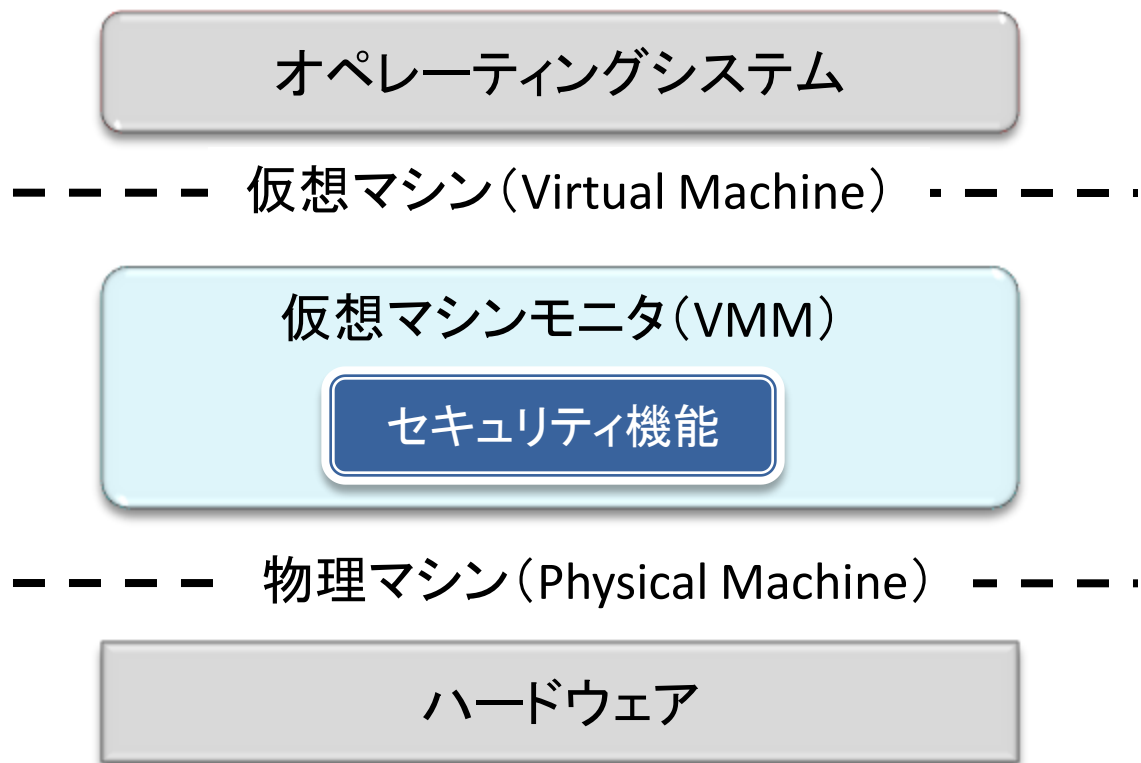
発表の流れ

- 1. セキュアVMとは(ユーザ向け)
 - 目的, 利用例, 脅威モデル
- 2. VMMアーキテクチャ概要
 - 設計方針, 基本アーキテクチャ
- 3. VMMアーキテクチャ詳細(開発者向け)
 - CPU, メモリ, デバイスの扱い

1. セキュアVMとは

セキュアVMとは

- 「セキュア」+「VM(仮想マシン)」
 - 仮想的にセキュアなマシン環境を提供する



「セキュア」とは

- 情報漏洩の防止 (Confidentiality)
 - PC・USBメモリの紛失・盗難
 - インターネットへの情報漏洩
 - ウィルスやファイル交換ソフトなど
- 情報漏洩事件の多発が背景
 - 月平均「**20件以上**」発生※
 - 2007年10月～12月だけで「68件」
 - 官公庁, 教育機関, 病院, 銀行, ...
 - エンドユーザからの情報漏洩が多い



The screenshot shows the 'Security NEXT' website with a table of personal information leakage incidents. The table has columns for date, category, and details. The title of the page is '個人情報漏洩事件一覧' (List of Personal Information Leakage Incidents).

日付	種別	概要
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。
2007/12/14	官公庁	国土交通省のウェブサイトから、国土交通省の職員の名前と住所が漏洩された。

※Security NEXT「個人情報漏洩事件一覧」より算出

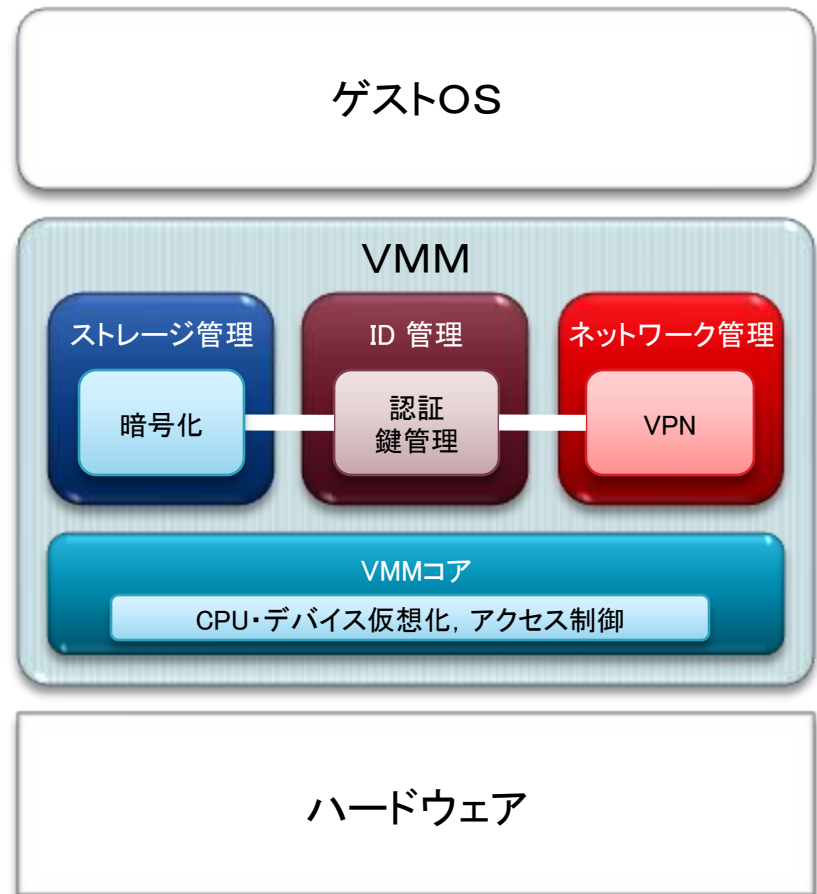
「VM」を使う目的

- 強力なセキュリティの実現
 - たとえOSが脆弱性でも大丈夫
- セキュリティの強制
 - 勝手にセキュリティを無効に出来ないようにする
 - 「ユーザ任せのセキュリティはやめたい」
(山口内閣官房情報セキュリティ補佐官談)

基盤となるコンピュータそのものをセキュアにしたい

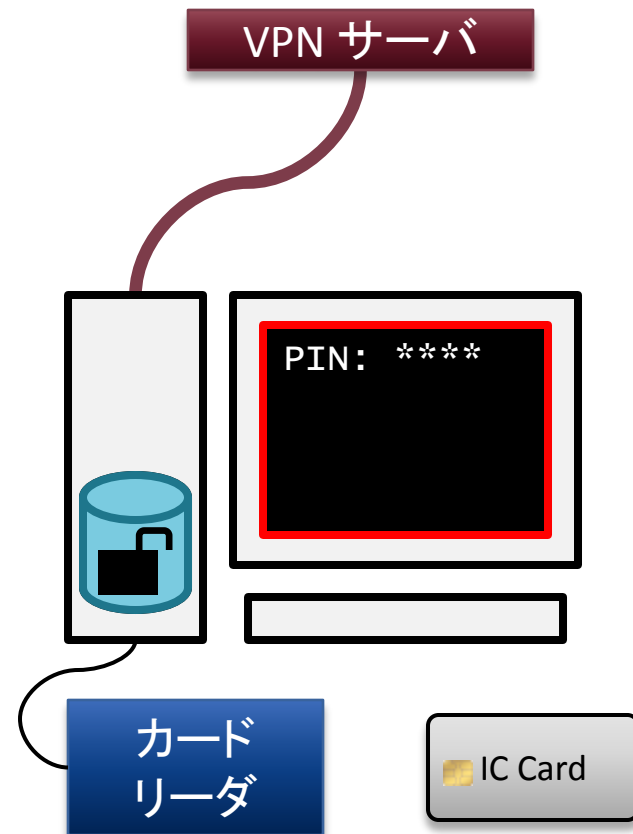
セキュアVMの機能構成

- ストレージ管理
 - HDD・USBメモリの暗号化
- ネットワーク管理
 - IPsecでVPN接続
- ID管理
 - ICカードで認証・鍵管理
- VMMコア
 - CPU・デバイスの仮想化
 - アクセス制御



セキュアVMの利用イメージ

- ログイン
 - ICカードを挿入する
 - PIN番号を入れる
- システムの起動
 - 暗号化が解除される
 - VPNが接続される
 - OSが起動する
- ログアウト
 - OSを停止する
 - ICカードを抜く



セキュアVMで防げる情報漏洩

- PCやメディアの紛失・盗難
 - HDDやUSBメモリはVMMで強制的に暗号化
 - 暗号鍵はICカードに格納
- ネットワーク経由での情報漏洩
 - VPNにより接続先がVMMで強制される
 - ネットワーク通信は自動的に暗号化される

対象としない情報漏洩

- 内部犯行
 - ICカードを持ったユーザによるハード的攻撃
 - 例: HDDを物理的に取り出してICカードの鍵で復号する
- 間接的手段
 - 画面を撮影する
 - 音声にデータを乗せて取り出す
 - 隠れチャネル

前半のまとめ

- 情報漏洩の防止
 - ストレージの暗号化
 - ネットワークの認証・暗号化
 - ICカードによるユーザ認証・鍵管理
- VMによるセキュリティの強制
 - OSに依存しない
 - ユーザに依存しない

2. VMMアーキテクチャ概要

検討事項

- 新規開発 v.s. 既存VMMの改良
 - Xen, QEMU など既存VMMを流用するか否か
- 完全仮想化 v.s. 準仮想化
 - ゲストOSに手を入れるか否か
- Type I v.s. Type II
 - ホストOSを使うか否か

制約条件 (MUST)

- WindowsXPが動作すること
 - ゲストOSに手を入れることは難しい
 - ⇒「完全仮想化」が必要
- 実運用を視野に入れること
 - 政府機関での使用, オープンソース公開
 - ⇒それなりの性能・完成度が必要
- 開発期間・コストは限定的である
 - 約2年半弱, 数億円
 - ⇒大がかりなシステムは作れない

前提条件 (SHOULD, MAY)

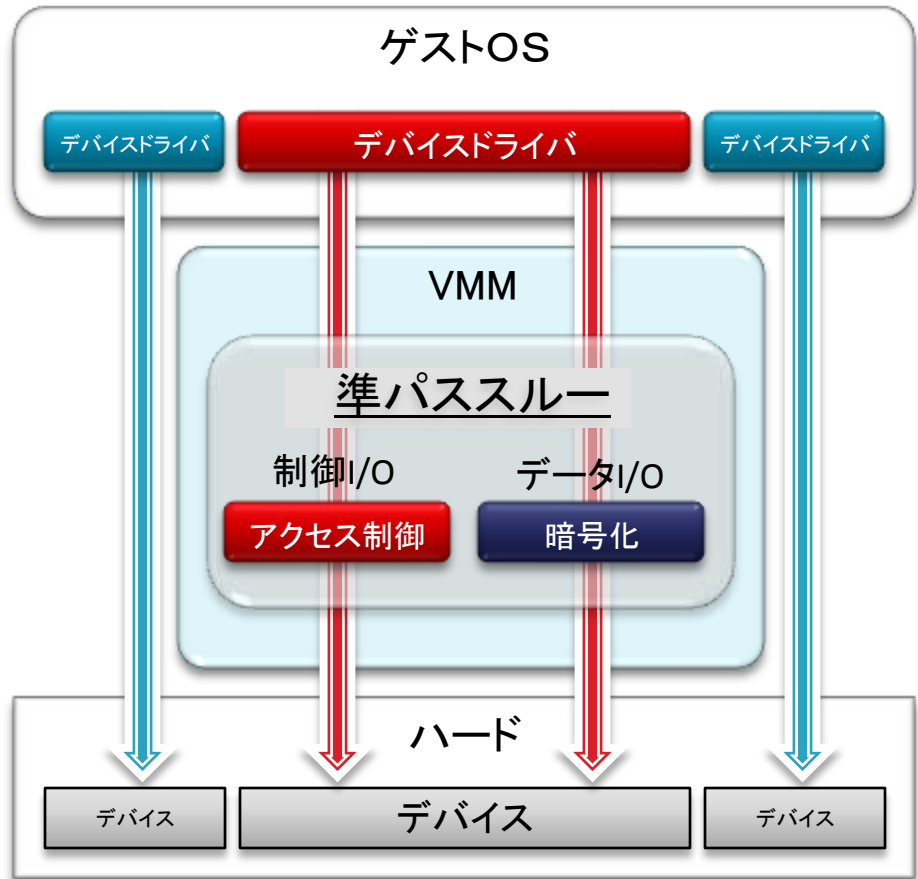
- VMM自身がセキュアである
 - ⇒VMMを出来るだけ小さくシンプルにすべき
- デスクトップ環境を想定する
 - ⇒サーバ統合のようなことはしなくてよい
- 対応デバイスは限定してよい
 - オフィス環境で必要なもののみ
 - 入札などで仕様を指定可能
 - ⇒独自のVMMを作ることが可能

設計方針

- 独自VMMを作成する
 - VMMを小さくシンプルに
 - Type I のVMM
 - 短期間・低コストでの実現
- 実運用に耐えうる性能・完成度
 - 「完全仮想化」する
 - デスクトップ用途

準パススルー型VMM

- 基本はパススルー
 - デバイスを仮想化しない
 - ゲストOSがデバイスを直接制御
- 必要最小限の監視・変換
 - 制御I/Oの監視
 - デバイスの状態把握
 - VMMに対するアクセス制御
 - データI/Oの変換
 - ストレージ・ネットワーク暗号化



準パススルー方式の利点

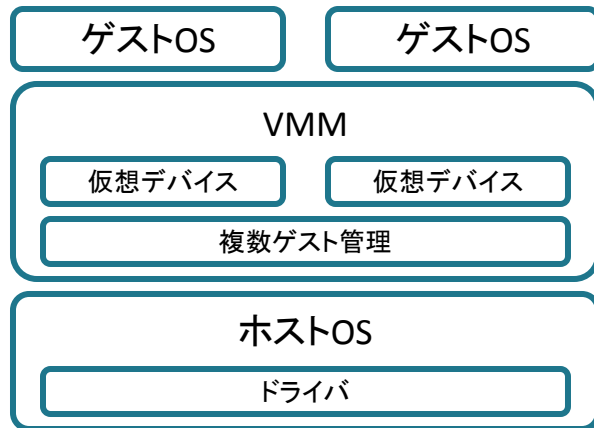
- セキュリティ向上
 - VMMを小さくできる
 - VMM自身の安全性が向上する(検証が容易)
- 性能・完成度
 - 仮想化のオーバーヘッドを大幅に削減できる
 - ゲストOSのデバイスドライバを活用できる
- 開発コストの削減
 - 0からの開発が現実的なコストで可能になる
 - デバイスドライバの数を限定できる

準パススルー方式の制限

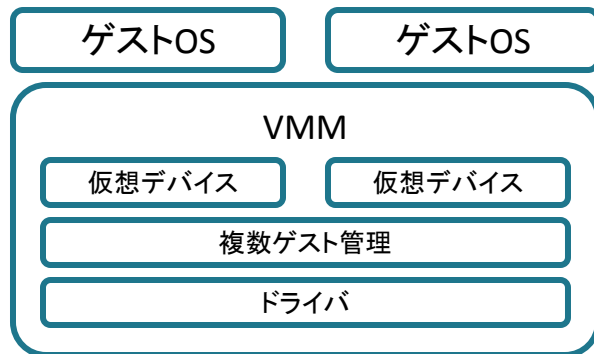
- 複数ゲストOSは同時に稼働しない
 - デスクトップ環境だから許容される
 - Windowsがセキュアな環境で動作すればよい
 - 将来的には対応できなくはない
 - 方法1:ハードウェアサポートを期待する
 - PCI SIGでIOV(I/O Virtualization)が策定中
 - 方法2:エミュレーション層を追加開発する
- 対応するマシン環境が限定される
 - 現状では許容されている
 - 今後の展開に期待？

アーキテクチャ比較

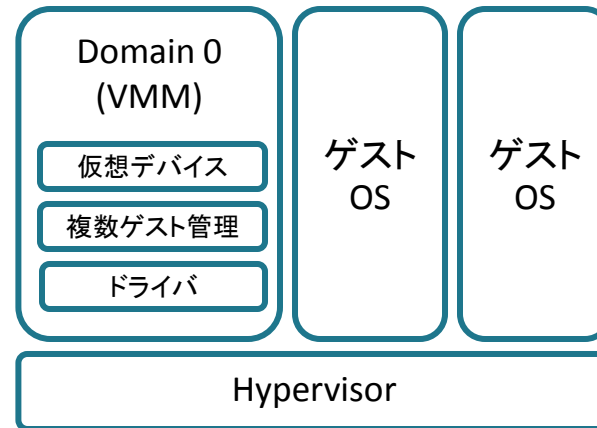
Type II 型 (VMWare Workstation等)



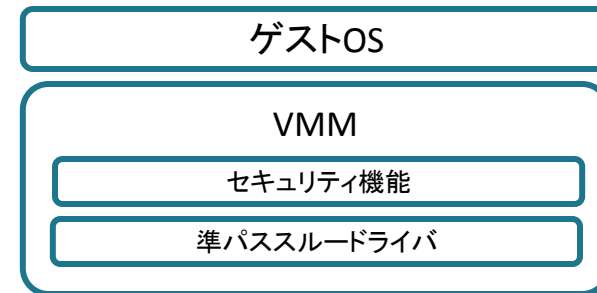
Type I 型 (VMWare Server等)



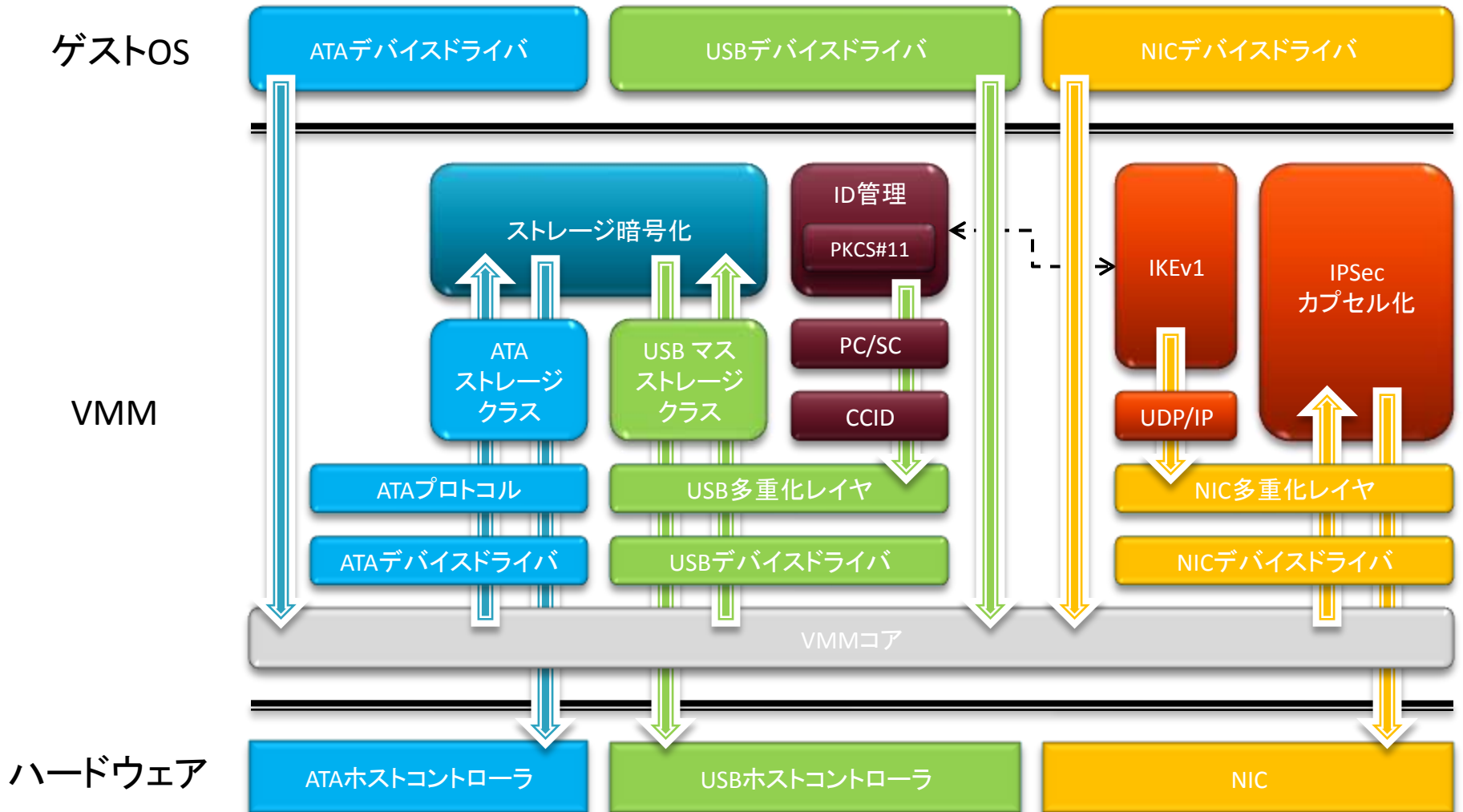
Para virtualization型 (Xen)



準パススルー型 (BitVisor)



VMMの構成



中盤のまとめ

- 準パススルー型アーキテクチャ
 - 可能な限りパススルー
 - 最小限必要なデバイスだけ監視・変換
- 独自VMMを0から作成
 - Type I 型 (ホストOSは使用しない)
 - 「完全仮想化」方式
 - ゲストOSは改変しない

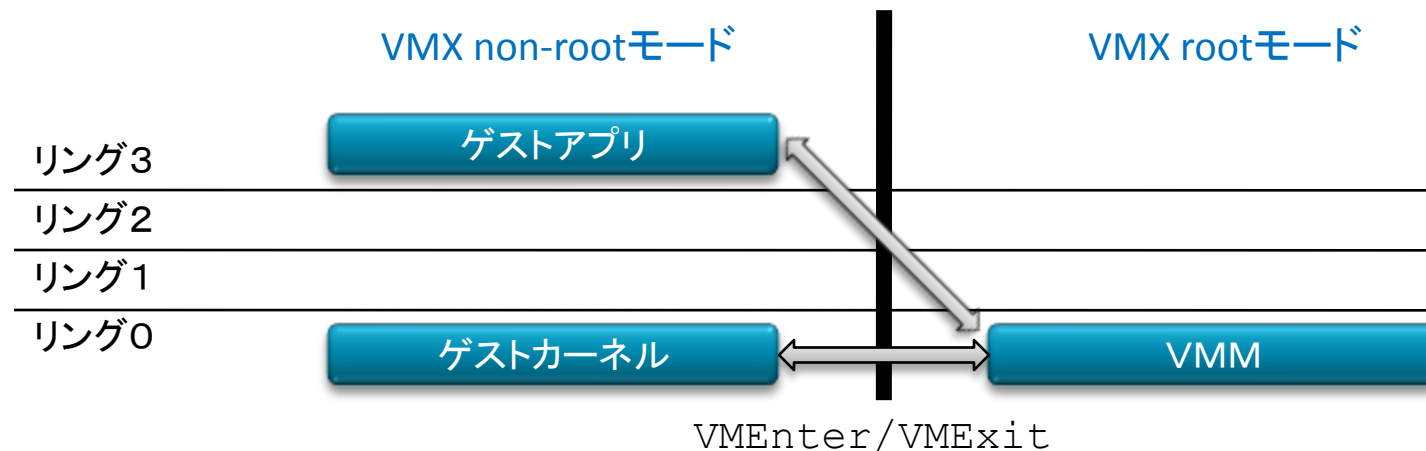
3. VMMアーキテクチャ詳細

準パススルー方式の実現

- CPUの仮想化
- メモリの仮想化
- デバイスの仮想化

CPUの仮想化

- Intel VT 機能を利用
 - VMX non-root モードと VMX root モード
 - VMCS(Virtual Machine Control Structure)領域
 - レジスタの状態などを保存するためのメモリ領域(4KB)



VMCS領域

Guest-state	VM-entry control	VM-execution control
Host-State	VM-exit control	VM-exit information

VMMで捉えるイベント

- I/O命令の一部
 - ATA, USB, NIC関連
- 特権命令の一部
 - 制御レジスタへの読み書きなど
- 割り込み・例外
 - ページフォルト関係など
- SMP関連

IN, INS, OUT, OUTS

MOV CRx

CPUID

RDMSR

WRMSR

INVLPG

Exceptions

NMI

External Interrupts

Interrupt window

INIT Signals

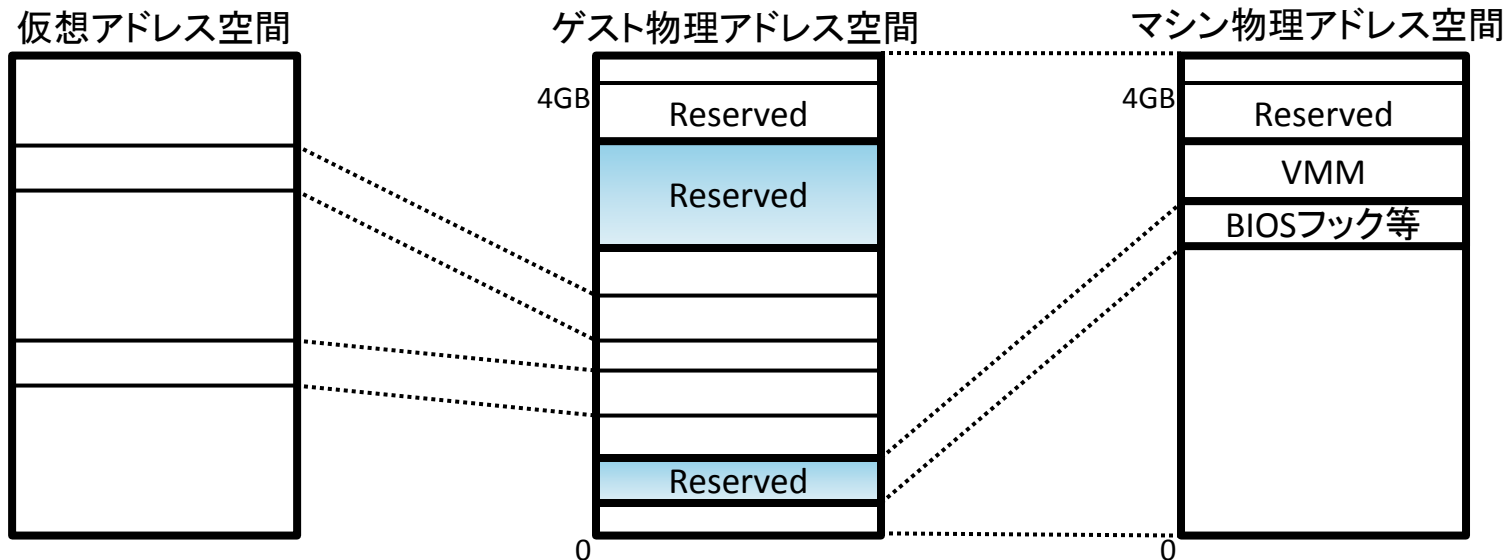
Start-up IPIs

VTで足りないこと

- リアルモード対応
 - 仮想8086 + 命令エミュレーションで対応
 - モード切替時, INS/OUTS命令, MMIO等でも利用
- メモリ管理
 - EPT(Extended Page Table)はまだない

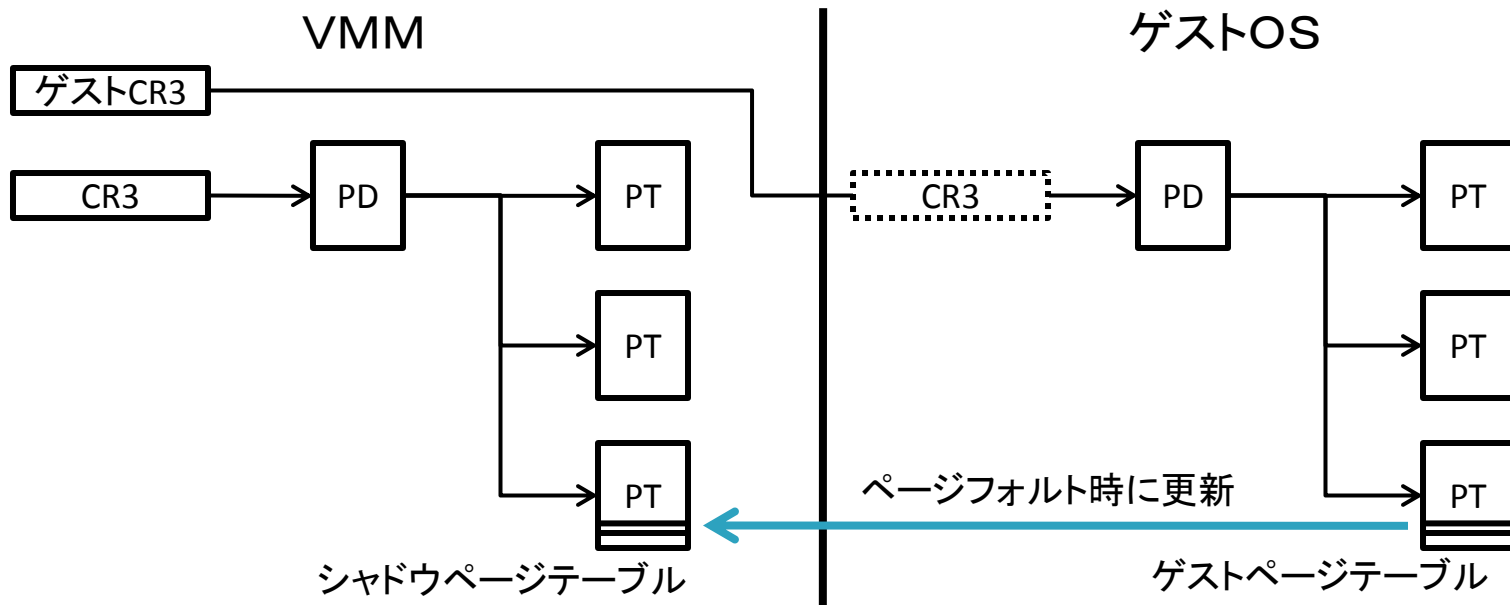
メモリの仮想化

- 基本的にはアドレス変換しない
 - ゲスト物理アドレス = マシン物理アドレス
- VMMは後端(<4GB)に常駐する
 - 現在のサイズは32MB
 - BIOSコール(e820h)をごまかしてreserved領域に



シャドウページング

- ゲストOSのシャドウ(コピー)をVMMで保持
 - VMM領域の保護
 - ページングオフ時の対応

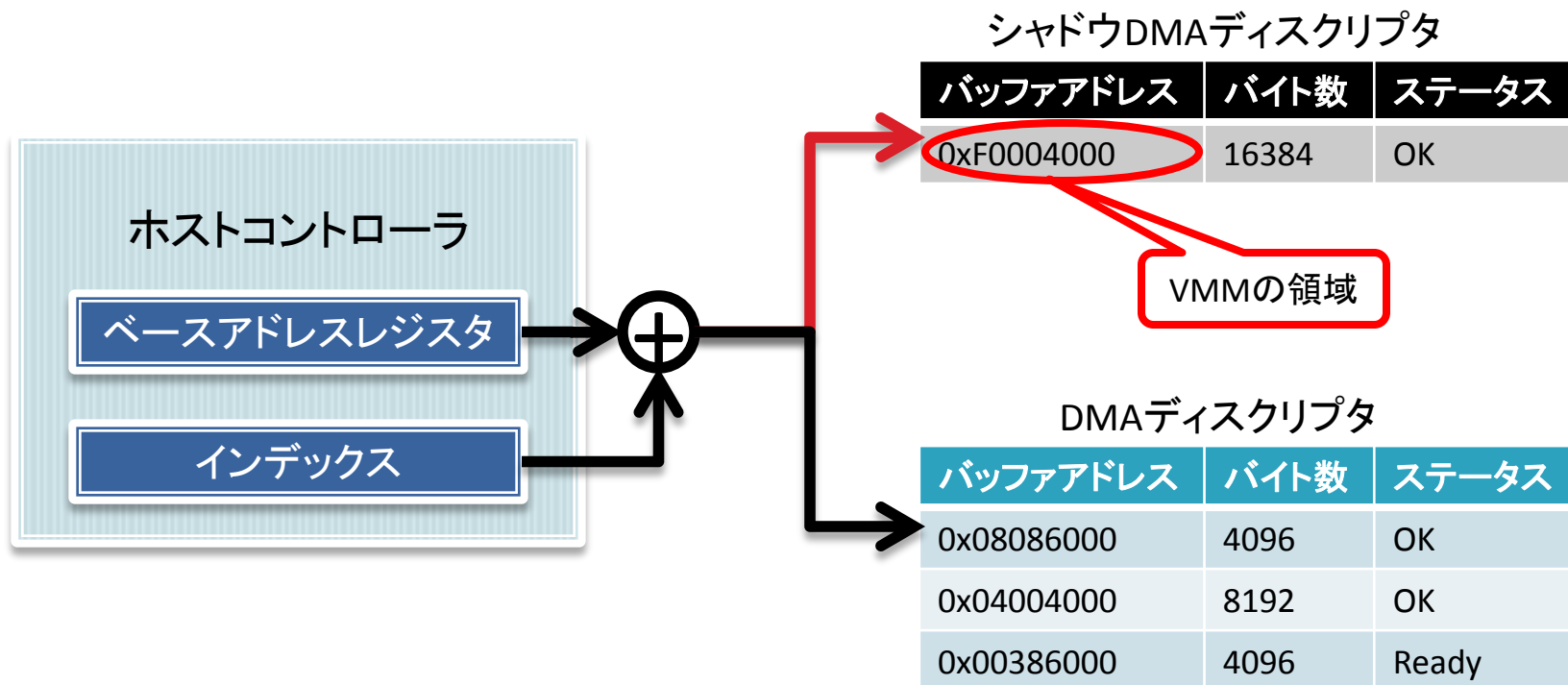


I/Oデバイスの仮想化

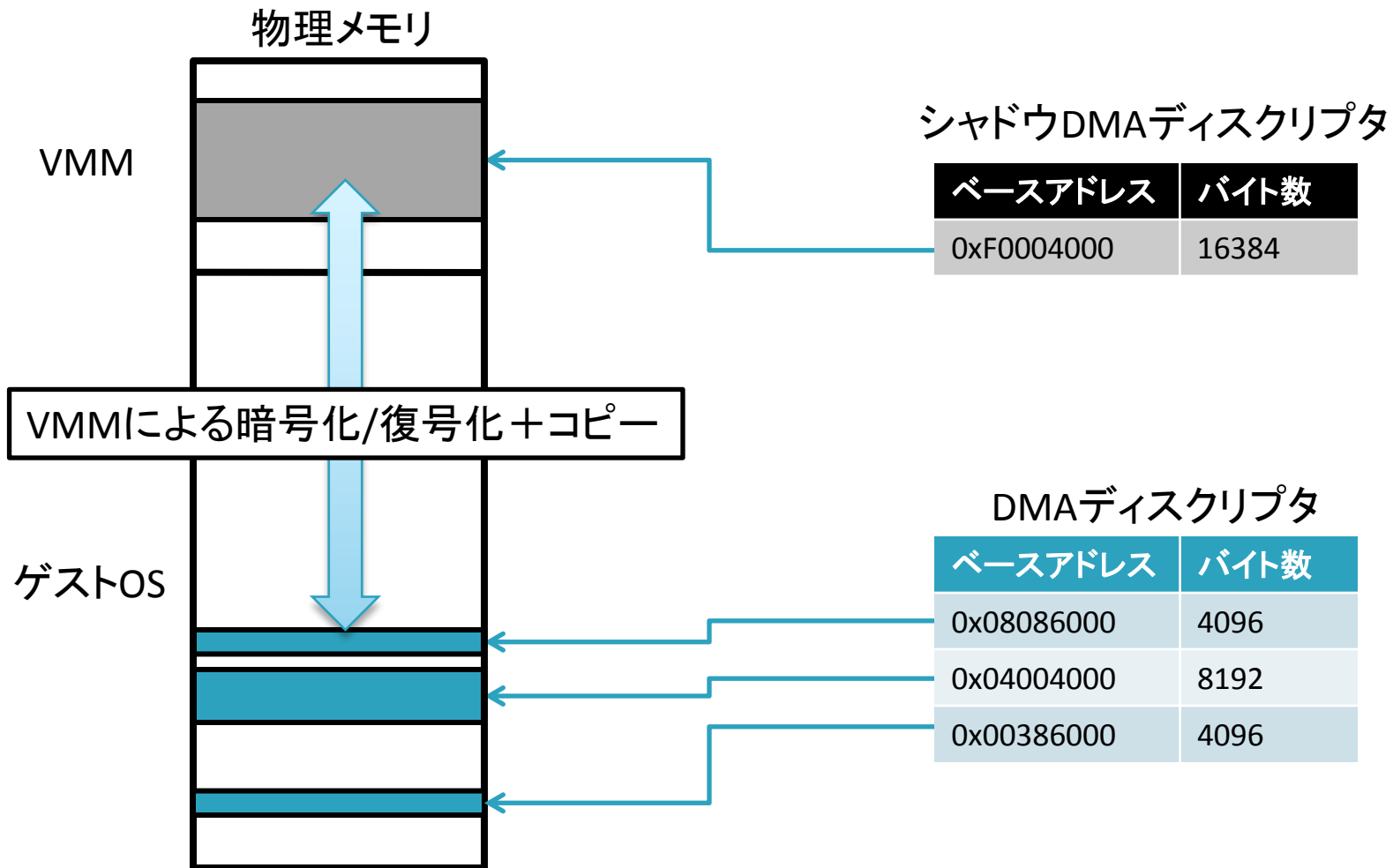
- I/O命令 (IN, OUTなど)
 - VTの機能によりVMMで捕捉
 - I/Oポートアドレス単位でビットマップで指定
 - 制御I/Oの監視, データI/Oの変換
- MMIO
 - シャドウページングにより捕捉
- DMA
 - VT-dによるアクセス制御
 - シャドウDMAディスクリプタ

シャドウDMAディスクリプタ

- DMAディスクリプタのシャドウを作る
 - データだけVMMで横取りする
 - 制御はゲストOSにさせる

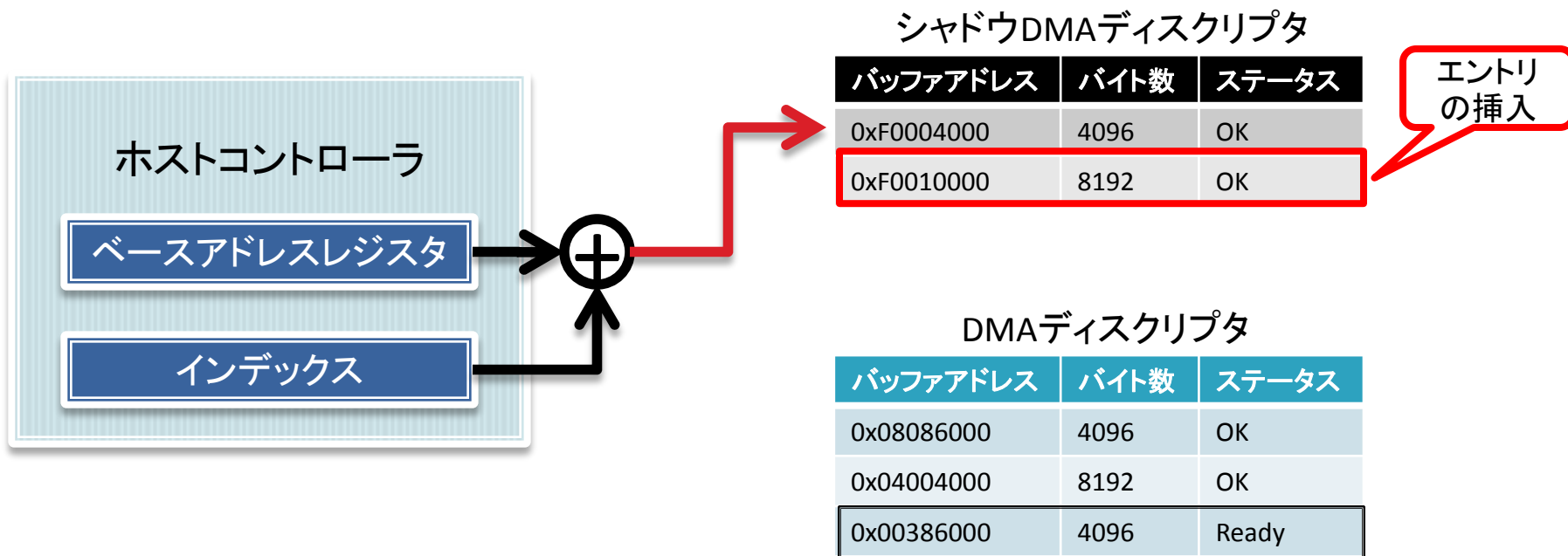


DMA転送データの暗号化



VMMによるデバイス利用

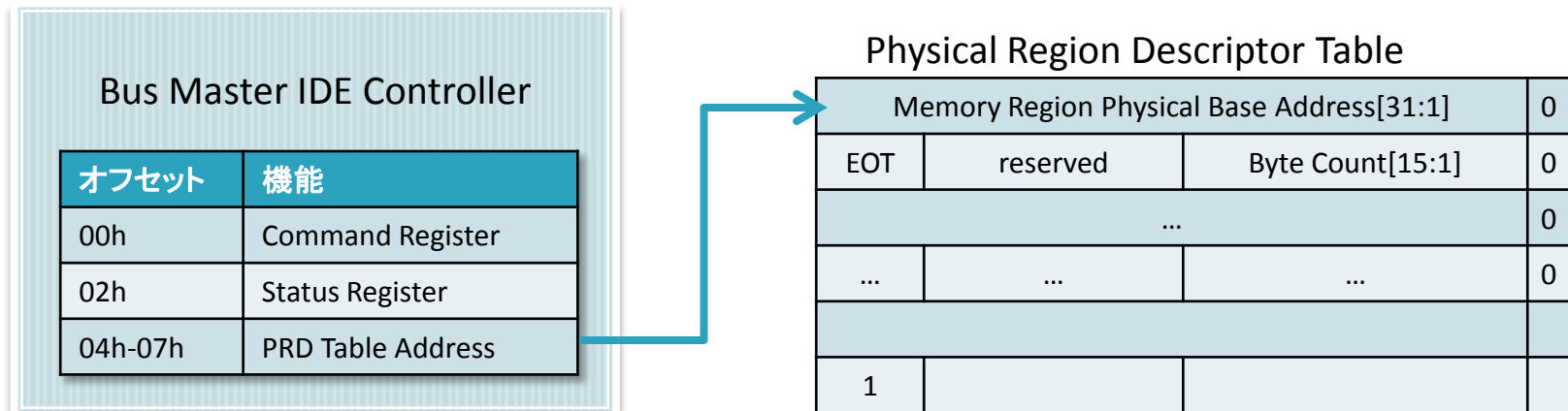
- VMMから一部のデバイスを使う必要がある
 - USB (ICカードリーダー), NIC (IKEプロトコル)
 - シャドウDMAディスクリプタの拡張で対応



ATAホストコントローラ

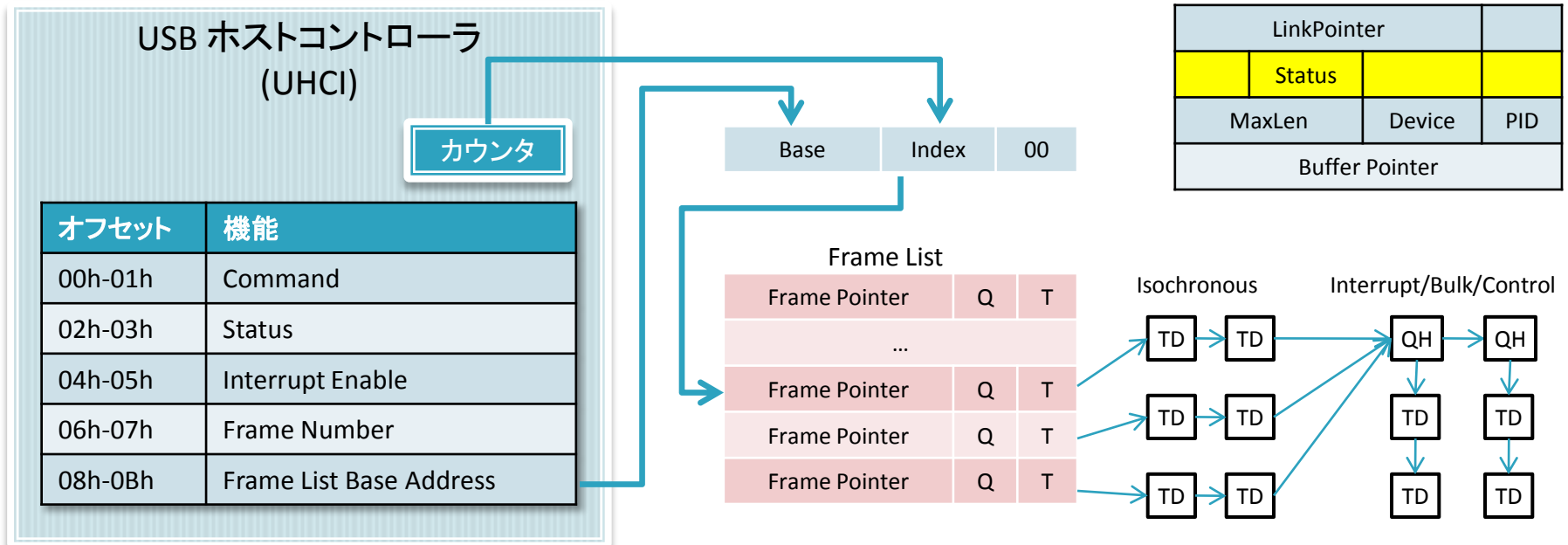
■ 比較的シンプルな構造

- PRD(Physical Region Descriptor)のテーブルをメモリ上に保持
- テーブルへのアドレスをレジスタに指定
- Start Bit に 1 を書き込むとDMA転送開始



USBホストコントローラ(UHCI)

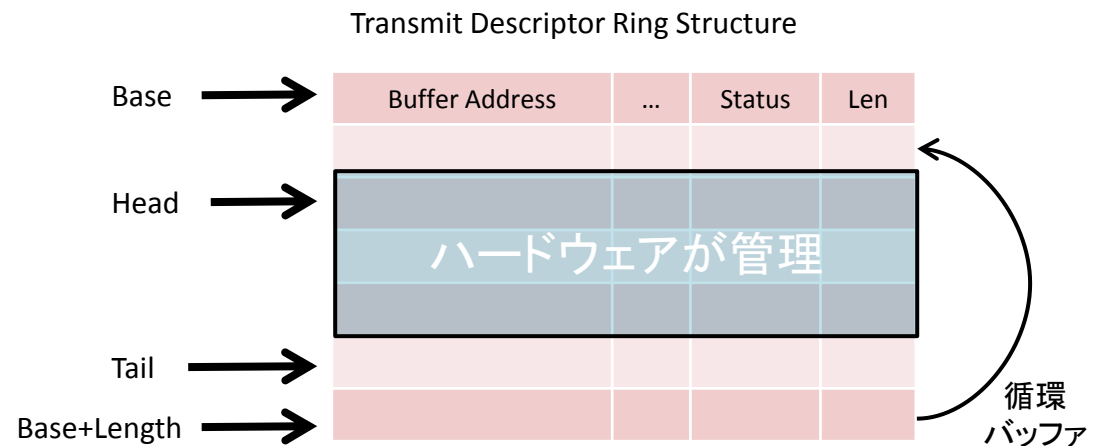
- 時間枠(1024ms)単位の転送
 - 1024エントリ × 4Byte=1ページのフレームリスト
 - TD(Transfer Descriptor)のリンクリスト



NIC(Intel PCI-E* GbE Controllers)

- ディスクリプタのリング構造
 - Tail ポインタをドライバソフトウェアで更新する
 - Head ポインタをホストコントローラが更新して追いかける
 - Tail に追いつくと自動的に停止する

NIC (Intel PCIe* GbE Controllers)	
オフセット	機能
03800h	Descriptor Base Low
03804h	Descriptor Base High
03808h	Descriptor Length
03810h	Descriptor Head
03818h	Descriptor Tail



- LVMM (Intel vPro)
 - クライアント端末のリモート管理が目的
 - ネットワークのみ仮想化し, 残りはパススルー
 - VMが2つある(User Partition と Service Partition)
- SecVisor [SOSP '07]
 - Linux カーネルの改竄防止
 - W⊕Xページ, 不正なカーネルモジュール防止
 - VMは1つ, デバイスは完全パススルー

公開スケジュール

- 平成19年度
 - VMMコア
 - ATAドライバ(IDE)
- 平成20年度
 - USBドライバ(UHCI:USB1.1)
 - USBドライバ(EHCI:USB2.0)
 - ATAドライバ(ATAPI) (AHCI未定)
 - NICドライバ(Intel PRO1000系)
 - ID管理
 - VPN管理
- 未対応デバイス
 - IEEE 1394
 - 無線LAN
 - プリンタ/スキャナ

全体のまとめ

- セキュアVMの概要
 - ストレージ暗号化
 - ネットワーク暗号化
 - ICカードによる認証・鍵管理
- 準パススルー型VMMの紹介
 - 可能な限りパススルー
 - 必要最小限のアクセスだけVMMで捕捉
- 準パススルー型の実装概要
 - Intel VT機能を活用
 - シャドウページテーブル
 - シャドウDMAディスクリプタ

国産VMM「BitVisor」

- 準パススルー型
- Type I 型 VMM (Hypervisor型)
- 64bit 対応 (ゲストOSは32bitのみ)
 - PAE対応
- マルチコア・マルチプロセッサ対応
- WindowsXP/Vista, Linux が動作
- コード行数約20,000行 (コアのみ)

本日ソースコード公開

BitVisor 0.2 (α版)

<http://www.securevm.org/>