Virtuelizacija

Intel VT-x

Virtuelizacija

- Nekoliko ključnih reči (podsećanje):
  - Virtuelni mašinski monitor (Virtual machine monitor – VMM) ili hipervizor
  - Gost (Guest) – proces kojem se pruža virtuelna kopija hardverskog sistema. Obično je ovo operativni sistem.
  - Domaćin (Host) – hardverski sistem na kome se sve izvršava.

- Uprošćen pogled na virtuelizaciju – VM se izvršavaju tako što VMM smenjuje VM-ove na procesoru. PC registaru se dodeli vrednost VM-va adresa trenutne instrukcije (PC) i VM započinje izvršavanje.

Virtuelizacija – problemi

- Šta se dešava, u sistemu bez virtualizacije, u sledećim situacijama?
  - Sistemski poziv
  - Pristup memoriji
  - I/O pristup

- U sistemu sa virtualizacijom, **virtuelna mašina (VM) treba da se ponaša kao ne virtuelizovana mašina.** Šta se onda dešava u prethodno spomenutim situacijama?

Virtuelizacija – problemi

Konkretnije:

- Ko i kako obrađuje prekid u gostu?

- Šta se dešava kada dve VM žele da pristupe istoj memorijskoj lokaciji?
  - Recimo VM A i VM B žele da pristupe lokaciji 1000h.

- Kako se pristupa I/O uređajima?

Virtuelizacija – problemi

- Ko i kako obrađuje prekid u gostu?
  - **Potrebno je obezbediti da Gost obrađuje izuzetke**.

- Šta se dešava kada dve VM žele da pristupe istoj memorijskoj lokaciji?
  - **Potrebno je virtuelizvoati memoriju**.

- Kako se pristupa I/O uređajima?
  - **Potrebno je virtuelizvoati I/O uređaje**.

Virtuelizacija – mod izvršavanja

- U kom modu treba da se izvršava gost?
  - **Ne u kernel modu**.
- Hardverska podrška (Intel VT-x, AMD V).

Obrada izuzetka – ne virtuelizovano okruženje

Proces	Hardver	OS
1. Izvršavanje instrukcije (add, load…)		
2. Sistemski poziv		
	3. Kernel mod (obrada izuzetka)	
		4. U kernel modu; Izvršavanje prekidne rutine
	5. Korisnički režim povratak u korisnički program	
6. Izvršavanje instrukcije (add, load…)		

Obrada izuzetka – u procesu gosta

- Obrada izuzetka je slična kao i u procesu koji nije virtuelizovan, ali VMM hvata izuzetke.
- VMM ne zna kako da obradi izuzetak ali zna gde se nalazi tabela prekidnih rutina gosta.
- VMM prilikom pokretanja gost OS-a ne dozvoljava gostu da inicijalizuje IVT (Intel x86 IDT), ali pamti gde se nalazi spomenuta tabela u memoriji kako bi kasnije mogla da pozove odgovarajuće prekidne rutine.

Obrada izuzetka – u procesu gosta

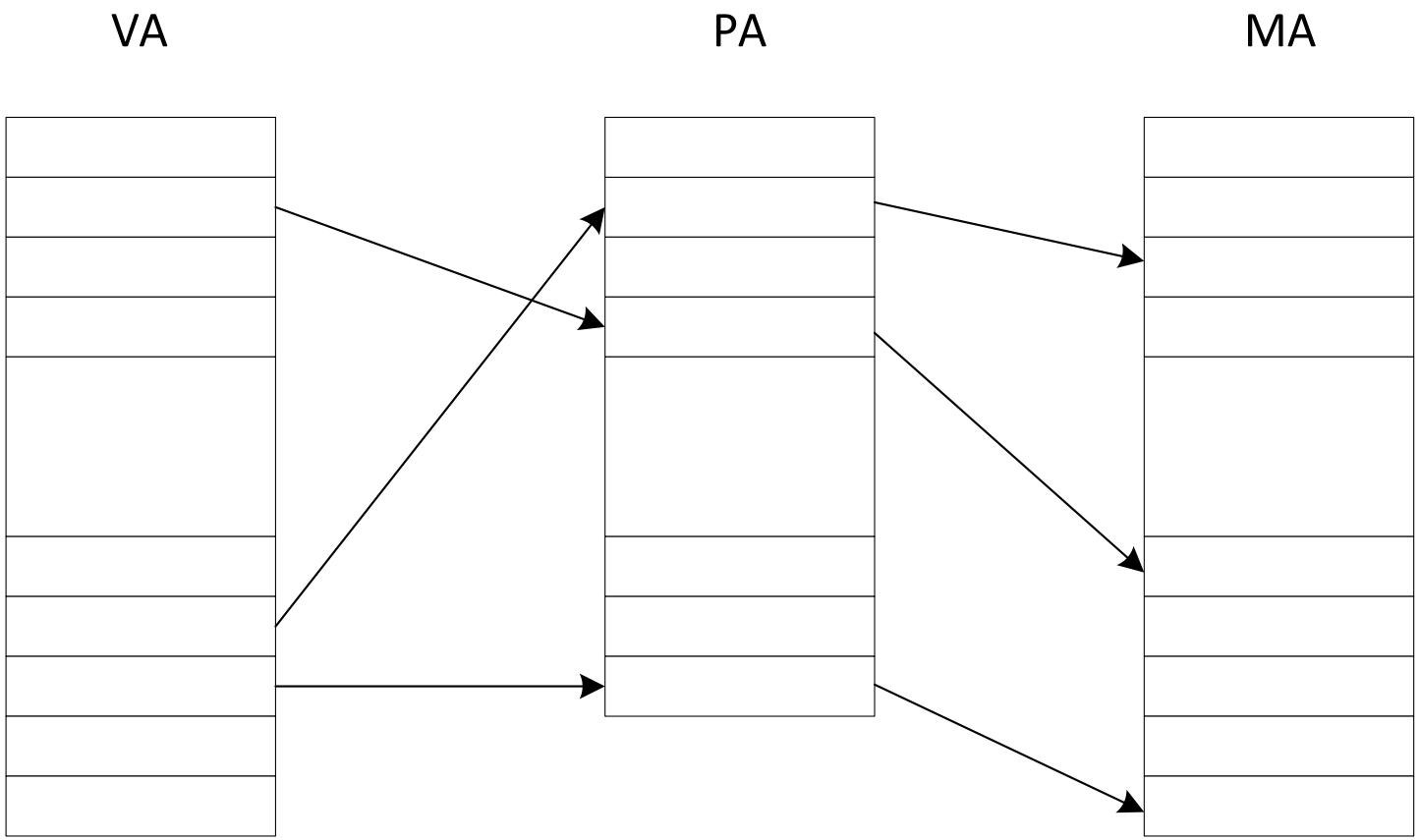
Proces	OS	VMM
1. Sistemski poziv		
		2. Pozovi gost OS prekidnu rutinu
	3. OS prekidna rutina: Izvršavanje i povratak iz P.R.	
		4. Vrati se u korisnički program
5. Izvršavanje instrukcije (add, load…)		

Virtuelizacija memorije

- U sistemu bez virtualizacije, procesi imaju svoj virtuelni adresni prostor koji se mapira u fizički adresni prostor.
- U slučaju sa virtualizacijom, mora da se doda još jedan nivo virtualizacije.
  - Šta ako dve VM žele da pristupe istoj lokaciji, npr. 1000h?

- VMM pravi razliku između realne i fizičke memorije (koje uglavnom označavaju isti pojam).
  - Virtuelna memorija
  - Realna memorija (fizička memorija)
  - Fizička memorija (mašinska memorija)

Virtuelizacija memorije



Virtuelizacija memorije

- VMM mora da vodi računa o preslikavanju realne memorije u fizičku memoriju.
- Jedno rešenje je softverska tehnika zasnovana na održavanju konzistentne verzije prateće tabele stranica tabele stranice gosta (shadow page table).

Virtuelizacija procesora (Intel)

- Operacije koje se koriste kao podrška za virtualizaciju su VMX operacije
- Postoje dve vrste operacija/moda:
  - VMX korene mod izvršavanja (VMX root operation) – VMM se ovde izvršava
  - VMX nekorene mod izvršavanja (VMX non-root operation) – VM se ovde izvršava
- Prelasci iz korenih operacija u ne korene operacije i obrnuto se naziva VMX tranzicija.
  - Tranzicija u VMX ne koreni mod se zove VM ulasci.
  - Tranzicija iz VMX ne koreni mod u koreni mod se zove VM izlasci.

Virtuelizacija procesora (Intel)

- Imamo četiri različite kombinacije modova izvršavanja:
  - root/system
  - root/user
  - non-root/system
  - non-root/user

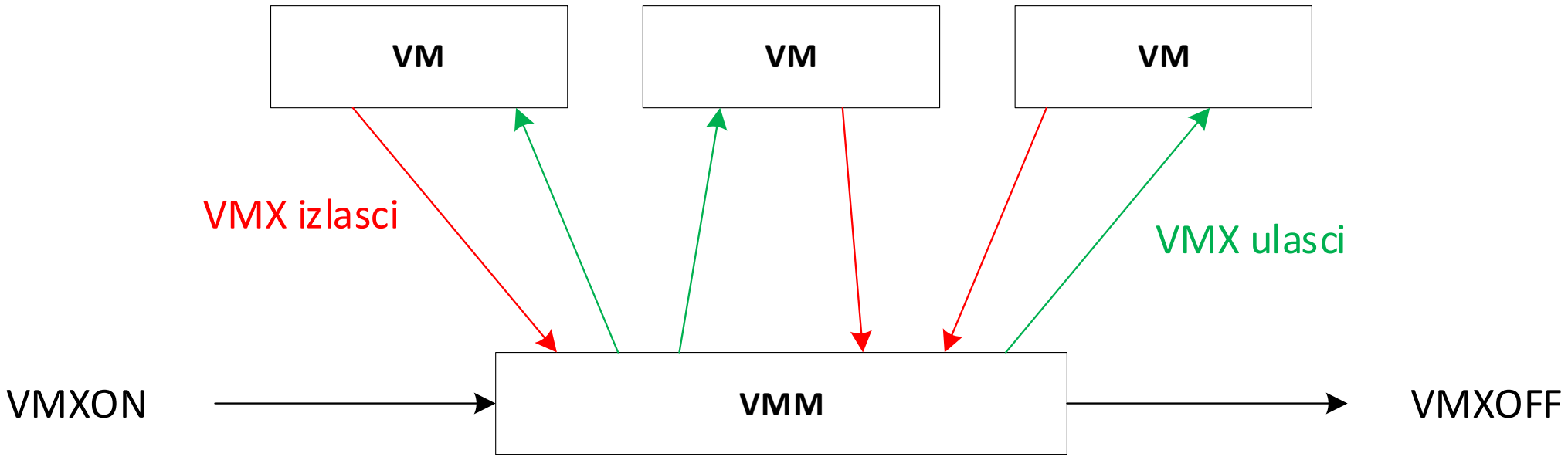
- Mod root/system ima više privilegija nego non-root/system.

Virtualizacija procesora (Intel)

- Procesor u VMX korenom modu:
  - Izvršavanje procesor u VMX korenom modu je prošireno sa VMX instrukcijama i upis u određene kontrolne registre je ograničen.

- Procesor u VMX ne korenom modu:
  - Izvršavanje procesor u VMX ne korenom modu je ograničen i modifikovan. Određene operacije i događaju dovode do toga da VM prekine izvršavanje i prepusti kontrolu VMM.
  - Na ovaj način je omogućeno VMM-u da kontroliše resurse procesora.

Životni ciklus VMM softvera



Životni ciklus VMM softvera

- Životni ciklus:
  - Softver ulazi u VMX mod izvršavajući **VMXON** instrukciju.
  - VMM može da pokrene gosta korišćenjem VM ulaznih instukcija: **VMLAUNCH** i **VMRESUME**
  - VMM može da povrati kontrolu izvršavajući VM izlazne instrukcije.
  - Gašenje VMM se postiže izvršavanjem **VMXOFF** instrukcije.

VM kontrolna struktura

- VMX ne koreni mod i VMX tranzicije se kontrolišu pomoću VM kontrolnih struktura (VMCS).
- VMM održava VMCS strukture. Uglavnom imamo jednu VMCS za svaki procesor VM-a.
- Pristup VMCS strukturama se kontroliše preko komponente procesora VMCS pokazivač. Postoji jedan pokazivač po logičkom procesoru.
- VMCS pokazivač je adresa VMCS-a.

VM kontrolna struktura - upravljanje

- Upravljanje VMCS:
  - Čitanje – VMREAD instrukcija
  - Pisanje – VMWRITE instrukcija
  - Brisanje sadržaja – VMCLEAR instrukcija

- Upravljanje VMCS pokazivačem:
  - Pisanje – VMPTRST instrukcija
  - Čitanje – VMPTRLD instrukcija

VM kontrolna struktura

- Sve spomenute instrukcije se izvršavaju nad trenutnom (current) VMCS strukturom.
- VMM vodi računa o svim VMCS strukturama, ali je samo jedna struktura trenutna nad kojom se izvršavaju instrukcije.

VM kontrolna struktura – polja strukture

- VMCS ima više polja:
  - Gost stanje – stanje procesora gosta.
  - Domaćin stanje – stanje procesora koje se koristi prilikom VMCS izlaza.
  - VM kontrola izvršavanja (VM execution control) – ovde možemo da naznačimo šta sme, a šta ne sme da se izvršava tokom non-root moda izvršavanja.
  - VM kontrola izvršavanja ulaza – flagovi koji određuju opciona ponašanja prilikom prelaska root u non-root mod izvšavanja.
  - VM kontrola izvršavanja izlaza – slično kao ulaz, samo što se ovo polje odnosi na prelaz iz non-root u root mod izvršavanja.
  - VM razlog izlaska – sadrži informacije o razlogu izvršavanja VM izlaska.

VM kontrolna struktura – polja strukture

- Gost stanje sadrži vrednosti različitih registara procesora, npr. registar koji sadrži pokazivač ka IV tabeli (%idtr registar), pokazivač ka trenutnom zadatku (current task; %tr registar), PC registar (instruction pointer) itd.
- Domaćin stanje sadrži vrednosti registara koje je potrebno učitati kada se izvrši VM izlaz. Ovo stanje sadrži i PC registar koji sad treba da ukazuje na rutinu koja obrađuje razlog izlaska iz VM.

VM kontrolna struktura – polja strukture

- VM kontrola izvršavanja sadrži (između ostalih):
  - Indikator (**Interrupt-Window Exiting flag**) koji omogućuje da gost OS obradi spoljašnje izuzetak bez napuštanja non-root moda.
  - Skup indikatora koji određuju da li će neka kritična instrukcija da izazove VM izlaz.
  - Po jedan indikator za svaki tip prekida; npr. da li će page fault da izazove VM izlaz ili ne.
  - Skup indikatora za IN/OUT instrukcije i za sve ostale I/O registre koji izazivaju VM izlaz.

VM kontrolna struktura – polja strukture

- VM razlog izlaska sadrži kod (spoljašnji prekid, I/O pristup itd.) koji označava razlog izlaska i nekoliko polja koja daju više informacija o razlogu izlaska.
- Polja u delu strukture za VM kontrolu izvršavanja ulaza i izlaza se odnose na prekide i na to kako se oni obrađuju.

Kako proveriti da li procesor podržava VMX?

- Proveriti podršku za VMX izvršavanjem CPUID.
- Ako je CPUID.1:ECX.VMX[bit 5] =1, procesor podržava VMX.

- CPUID.1:ECX.VMX[bit 5] =1 označava da je potrebno inicijalizovati registar EAX na 1, pa zatim pozvati instrukciju CPUID i nakon izvršavanja instrukcije proveriti sadržaj registra ECX, konkretno bit 5 koji se odnosi na VMX.

Kako dozvoliti korišćenje VMX?

- Potrebno je postaviti CR4.VMXE[bit 13] na 1. Nakon postavljanja spomenutog bita na 1, VMXON operacija može da se izvrši.
- Ovaj bit jedino može da se postavi na 0 van VMX moda rada. Nakon VMXOFF instrukcije.

- Izvršavanje VMXON instrukcije je takođe uslovljeno MSR registrom IA32_FEATURE_CONTROL MSR (adresa 3Ah). Relevantni bitovi ovog regista:
  - Bit 0 – daje opciju da se u BIOS-u podesi podrška za VMX.
  - Bit 1 – dozvoljava VMXON operaciju u SMX moda rada.
  - Bit 2 – dozvoljava VMXON operaciju van SMX moda rada.

Kako dozvoliti korišćenje VMX?

- Pre izvršavanja VMXON instrukcije, potrebno je alocirati 4KB poravnat region memorije koji logički procesori koriste kao podršku za VMX operacije. Ovaj region memorije se zove **VMXON region**.
- Količina memorije koja se alocira će se možda promeniti u budućnosti. Što nije problem, jer VMX je napravljen tako da bude proširiv.