AMD64 RMPOPT

Publication # **69201**	Revision: **1.00**
Issue Date: **February 2026**	

Future AMD APM releases that describe this feature supersede the information in this document

Revision History

Date	Revision	Change Description
February 2026	1.00	Initial release

RMPOPT

Introduction

The RMPOPT feature minimizes the performance overhead of RMP checks for the hypervisor and non-SEV-SNP guests. In the SEV-SNP architecture, the hypervisor and non-SEV-SNP guests are subject to RMP checks on writes to protect against corruption of SEV-SNP guest memory. The RMPOPT architecture enables optimizations by which these checks can be skipped if large 1GB regions of memory are known to not contain any SEV-SNP guest memory.

Presence

CPUID Fn8000_0025_EDX[Rmpopt] (bit 0) indicates RMPOPT support.

RMPOPT MSR

The RMPOPT feature is enabled through the per-core RMPOPT_BASE MSR (C001_0139h).

RMPOPT Base MSR Layout

Field Name	Bit Position	Access	Description
RmpoptEn	0	R/W	RMPOPT feature enable
RmpoptTableSize	22:1	RO	Maximum address space size supported by the processor for RMPOPT expressed in Gbytes
Reserved	29:23	R/W	MBZ
RmpoptBaseAddr	51:30	R/W	Start address of the memory covered by the CPUs RMPOPT table expressed in Gbytes
Reserved	63:52	R/W	MBZ

The RmpoptEn bit may only be written to 1 if SYSCFG[SNPE] and SEGMENTED_RMP_CFG[SegRmpEn] are 1. The RmpoptEn bit cannot be cleared to 0 when SYSCFG[SNPE] is 1. The RmpoptBaseAddr field is read-only when RmpoptEn is 1. An attempt to write RMPOPT Base MSR when requirements are not met will result in a #GP(0) exception.

RMPOPT Table

Each core has the RMPOPT Table which indicates if specific 1Gbyte system memory regions are entirely hypervisor-owned. Each 1Gbyte of the system address space starting at RmpoptBaseAddr is represented by a single bit, indicating if the entire region is hypervisor-owned.

Software may choose to optimize different regions of the system address space for different cores. The combination of RmpoptBaseAddr and RmpoptTableSize describes which portion of the system address space may be optimized for each core.

When performing memory accesses other than to private memory by an SEV-SNP guest, the processor may consult the RMPOPT table to potentially skip RMP access. If the address of the memory access resides within an address region covered by the local RMPOPT table, and the table indicates the entire region is hypervisor-owned, then the RMP check may be skipped, thus improving performance. If the address of the memory access does not reside within the memory covered by the local RMPOPT table, or if the table indicates the region may not be entirely hypervisor-owned, RMP checks must be performed on write requests.

The RMPOPT table is managed by a combination of software and hardware. Software uses the RMPOPT instruction to set bits in the table, indicating that regions of memory are entirely hypervisor-owned. Hardware automatically clears bits in the RMPOPT table when RMP contents are changed during RMPUPDATE instruction.

RMPOPT Instruction

The RMPOPT instruction checks if a region of memory is entirely hypervisor-owned and updates the RMPOPT table. RMPOPT takes two inputs, a starting system physical address in RAX, rounded down to the nearest Gbyte, and the operation type in RCX, and returns operation specific information in RFLAGS.CF.

If RCX is equal to 0, the processor verifies if the entire 1 Gbyte region starting at the provided system physical address is hypervisor-owned, updates the RMPOPT table and indicates to software if the optimization was successful in RFLAGS.CF. RFLAGS.CF value of 1 means that the region is optimized.

If RCX is equal to 1, the processor returns the optimization status for the 1Gbyte region in RFLAGS.CF. RFLAGS.CF value of 1 means that the region is optimized.

This is a privileged instruction. Attempted execution at a privilege level other than CPL0 will result in a #GP(0) exception. In addition, this instruction is only valid in 64-bit mode with RMPOPT enabled; in all other modes a #UD exception will be generated.

The RMPOPT intercept is VMCB offset 14h, bit 7.

Mnemonic	Opcode	Description
RMPOPT	F2 0F 01 FC	Manages RMPOPT table

rFlags Affected

ID	VIP	VIF	AC	VM	RF	NT	IOPL	OF	DF	IF	TF	SF	ZF	AF	PF	CF
																M
21	20	19	18	17	16	14	13:12	11	10	9	8	7	6	4	2	0

Exceptions

Exception	Real	Virtual 8086	Protected	Cause of Exception
Invalid opcode, #UD	X	X	X	Instruction not supported as indicated by CPUID Fn80000025_EDX[Rmpopt] = 0
	X	X	X	This instruction is only recognized in 64-bit mode.
			X	Instruction is not enabled MSRC001_0139[RmpoptEn] = 0

Exception	Real	Virtual 8086	Protected	Cause of Exception
General Protection, #GP			X	CPL was not 0