Microarchitectural Data Sampling (MDS) FAQ for VxWorks

CVE-2018-12126, CVE-2018-12127,CVE-2018-12130,CVE-2019-11091, also referred to as Microarchitectural Data Sampling (MDS), are security vulnerabilities that potentially allow for the unauthorized gathering of sensitive data from computing devices.

These exploits are based on side-channel analysis. A side-channel is some observable aspect of a computer system's physical operation, such as timing, power consumption or even sound. The statistical analysis of these behaviors can in some cases be used to potentially expose sensitive data on computer systems that are operating as designed. These exploits do not have the potential to corrupt, modify or delete data. All of the methods take advantage of speculative execution, a common technique in processors used to achieve high performance.

Is my product impacted?

How do I determine whether the prerequisites are met?

Is this a bug in VxWorks software?

How can I continue to stay updated?

Is my product impacted?

For a product to be vulnerable to MDS, four prerequisites must be in place:

1. The processor must be affected by MDS
2. The attacker must have an available timing source granular enough to measure the impacts of speculative execution
3. The attacker must be able to run attack code on the processor in one memory domain, e.g. a user space process or a VM
4. There must be information accessible to the processor, that an attacker is not authorized to access, in another memory domain than the one where the attach code is executed,

If all four prerequisites are met, then the product is potentially vulnerable to one or more of the MDS exploits.

How do I determine whether the prerequisites are met?

1. The processor must implement features that can be exploited by MDS
    a. Determine whether your processor is vulnerable to MDS using the processor vendor's guidance and documentation
2. The attacker must have an available timing source granular enough to measure the impacts of speculative execution
    a. This can be somewhat system dependent, but in general if only coarse grain timers are available (i.e., smallest timing measurement is on the order of 1000 processor clock cycles or more), exploiting timing of speculative execution is extremely challenging
3. The attacker must be able to run attack code on the processor
    a. MDS vulnerabilities are not remote attacks, they require attacker code to be running on the processor and measuring the timing of speculative execution. Many embedded systems or mission critical systems are highly constrained and only allow a prescribed, pre-defined, and verified set of functions to run on the system. In these types of systems, for example, the likelihood of an attacker running attack code on the processor may be low enough that the risk is acceptable.
4. There must be information accessible to the processor in another memory domain than the one where the attach code is executed, that an attacker is not authorized to access
    a. MDS issues do not have the potential to corrupt, modify, or delete data, so if the system does not contain information that an attacker is not authorized to access, there would be no impact related to a successful attack and the risk may be acceptable. For example, in platforms that do not implement separation between privilege modes (e.g., kernel vs. userspace), any code running on the processor is authorized to access any data available to the processor.

If the prerequisites are met, and it is determined that the product is potentially vulnerable to one or more of the MDS exploits, Wind River encourages our customers to take a risk-based approach to determining the appropriate solutions for their system. These solutions may include the processor vendor recommended mitigations, and/or other alternatives that mitigate the risk associated with MDS exploits. Examples of alternative mitigations include:

- Integration of secure boot and transition to a closed environment that prevents attackers from being able to run exploit code on the processor
- Storage of, and operation on, information that needs to be protected in a hardware security module such as a TPM so that sensitive information is not available in processor memory where it could be disclosed through a MDS exploit

Is this a bug in VxWorks software?

No. This is not a bug or a flaw in VxWorks. These new exploits leverage data regarding the proper operation of processing techniques common to modern computing platforms, potentially compromising security even though a system is operating exactly as it is designed to.

How can I continue to stay updated?

In general, additional information about any and all updates to Wind River products is available on Wind River's product support site - https://knowledge.windriver.com.

References

https://software.intel.com/security-software-guidance/software-guidance/microarchitectural-data-sampling

https://software.intel.com/security-software-guidance/insights/deep-dive-intel-analysis-microarchitectural-data-sampling