Plundering of crypto keys from ultrasecure SGX sends Intel scrambling again

reader comments

64 with 49 posters participating, including story author

Share this story

For the past two years, modern CPUs—particularly those made by Intel—have been under siege by an unending series of attacks that make it possible for highly skilled attackers to pluck passwords, encryption keys, and other secrets out of silicon-resident memory. On Tuesday, two separate academic teams disclosed two new and distinctive exploits that pierce Intel's Software Guard eXtension, by far the most sensitive region of the company's processors.

Abbreviated as SGX, the protection is designed to provide a Fort Knox of sorts for the safekeeping of encryption keys and other sensitive data even when the operating system or a virtual machine running on top is badly and maliciously compromised. SGX works by creating trusted execution environments that protect sensitive code and the data it works with from monitoring or tampering by anything else on the system.

Key to the security and authenticity assurances of SGX is its creation of what are called enclaves, or blocks of secure memory. Enclave contents are encrypted before they leave the processor and are written in RAM. They are decrypted only

after they return. The job of SGX is to safeguard the enclave memory and block access to its contents by anything other than the trusted part of the CPU.

Raiding Fort Knox

Tuesday’s attacks aren’t the first to defeat SGX. In 2018, a different team of researchers broke into the fortified Intel region after building on an attack known as Meltdown, which, along with a similar attack known as Spectre, ushered in the flurry of processor exploits. A different team of researchers broke SGX earlier this year.

Intel mitigated the earlier SGX vulnerability by introducing microcode updates. However, these mitigations did not last, as two new attacks have sent Intel scrambling anew to devise new defenses. Intel released the new updates on Tuesday and expects them to be available to end users in the coming weeks. Depending on the computer, the fix will either be installed automatically or will require manual intervention. Users, particularly those who rely on the SGX, should check with the manufacturer of their machine and ensure that the update is installed as soon as practical.

The new SGX attacks are known as SGAxe and CrossTalk. Both break into the fortified CPU region using separate side-channel attacks, a class of hack that infers sensitive data by measuring timing differences, power consumption, electromagnetic radiation, sound, or other information from the systems that store it. The assumptions for both attacks are roughly the same. An attacker has already broken the security of the target machine through a software exploit or a malicious virtual machine that compromises the integrity of the system. While that’s a tall bar, it’s precisely the scenario that SGX is supposed to defend against.

Stealing attacker-chosen secrets

SGAxe is able to steal large chunks of SGX-protected data of an attacker’s choice. One class of sensitive data is that belonging to the target user—for instance, wallet addresses or other secrets used in financial transactions involving blockchains. The picture on the left immediately below this paragraph shows an image file that was stored in a secure enclave. The one on the right shows the same image after it was extracted using SGAxe.

van Schaik et al.
The attack can just as easily steal cryptographic keys that SGX uses for “attestation,” or the process of proving to a remote server that the hardware is a genuine Intel processor and not a malicious simulation of one. A remote server can require connecting devices to provide these attestation keys before it will carry out financial transactions, play protected videos, or perform other restricted functions. In a

paper titled SGAxe: How SGX Fails in Practice, researchers from the University of Michigan and the University of Adelaide in Australia wrote:

> *With the machine's production attestation keys compromised, any secrets provided by [the] server are immediately readable by the client's untrusted host application while all outputs allegedly produced by enclaves running on the client cannot be trusted for correctness. This effectively renders SGX-based DRM applications useless, as any provisioned secret can be trivially recovered. Finally, our ability to fully pass remote attestation also precludes the ability to trust any SGX-based secure remote computation protocols.*

https://arstechnica.com/?p=1682405