Name:Rahman,Md Mojibur

Id: 15-30527-3

1.What is meltdown and spectre?

->Meltdown and Spectre exploit critical vulnerabilities in modern

processors. These hardware vulnerabilities allow programs to steal

data which is currently processed on the computer. While programs

are typically not permitted to read data from other programs, a

malicious program can exploit Meltdown and Spectre to get hold of

secrets stored in the memory of other running programs.Meltdown

breaks the most fundamental isolation between user applications and

the operating system. This attack allows a program to access the

memory,and thus also the secrets,of other programs and the operating

system.

2. how to protect it?

->First,and most important:Update operating system as soon as possible.

The more severe flaw, Meltdown, affects “effectively every [Intel]

processor since 1995,” according to the Google security researchers

that discovered it. It’s an issue with the hardware itself, but the

major operating system makers have rolled out updates that protect

against the Meltdown and Spectre CPU flaws.

Microsoft pushed out an emergency Windows patch late in the day on

January 3. If it didn’t automatically update your PC, head to Start

> Settings > Update & Security > Windows Update, then click the Check

now button under “Update status.”

The system should detect the available update and begin downloading

it. Install the update immediately. We do not recommend manually

installing the Windows Meltdown patches if Microsoft hasn’t pushed

them to PC via Windows Update.

Also need to protect against Spectre, which tricks software into

accessing your protected kernel memory. Intel, AMD, and ARM chips are

vulnerable to Spectre to some degree. Software applications need to

be updated to protect against Spectre.

3. how they work?

->Spectre and Meltdown exploit the same flaw in the hardware but they

use different routes to gather the information. What happens in both

cases is that the processor is made to execute instructions it should

never execute as part of its speculative execution hardware. Once the

processor discovers that it shouldn't have carried out the instructions

it removes all trace that the instructions were ever carried out - except

of course it forgets to reset the cache back to its original state. It

doesn't cache the data that should never have been accessed, but if that

data is used to access some other data used as an address then that data

is still in the cache. To find the data that is in the cache just needs a

comparison of access times to reveal which of a possible set of data has

been acessed. Which data have been cached gives you the value of the

|restricted data that you should never have had access to.

The Meltdown exploit uses an attempted access to the OS kernel address space

to fire an exception, but not before speculative execution has retrieved and

used the off-limits data and left a trace of it in the cache. Meltdown is an

exploit directed at the kernel and as such is fairly easy to counter by keeping

kernel and user addresses separate and this is the basis of the fix that is

being rolled out for most operating systems at the moment.