What is hardware hacking? Is it worrisome?

Hardware hacking involves exploiting vulnerabilities in the physical components of a device. Unlike software hacking, attackers must be on-site and need physical - and reasonably uninterrupted - access to the target device to perform a hardware hack.

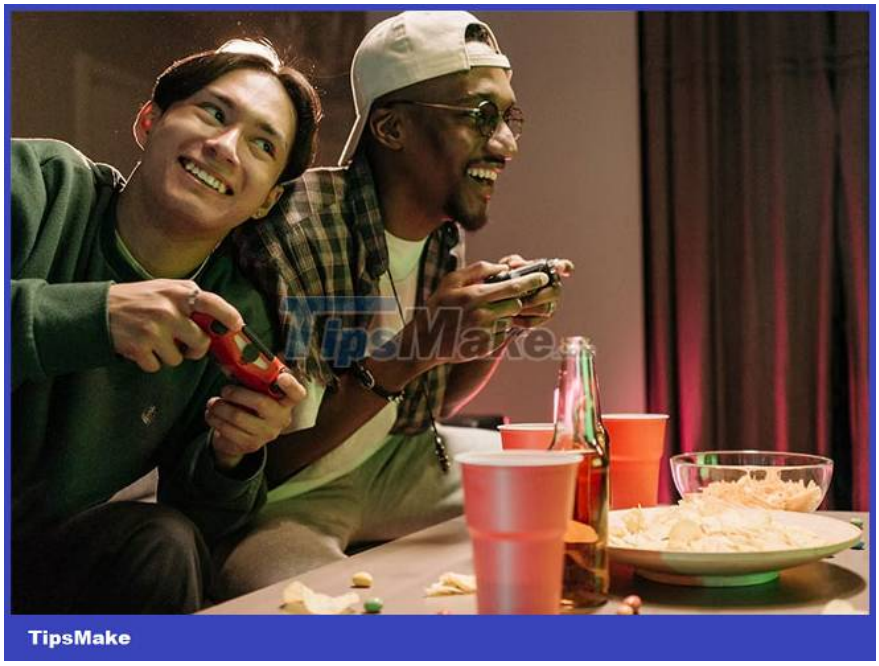
Software is the first thing that comes to mind when you hear that someone, company or organization has been hacked. This is understandable since software is the 'brain' of modern devices. So control software gives attackers the power to lock users, steal data, or wreak havoc. Accessing software is also easier because attackers don't need to be near their target. However, software updates can keep hackers out, and companies have become adept at stopping attacks and fixing any vulnerabilities. Software security also makes it more cost-effective.

Hardware security, however, is another story.

What exactly is hardware hacking?

Hardware hacking involves exploiting vulnerabilities in the physical components of a device. Unlike software hacking, attackers must be on-site and need physical - and reasonably uninterrupted - access to the target device to perform a hardware hack. The tools needed to compromise a device can be hardware, software, or a combination of both, depending on the target.

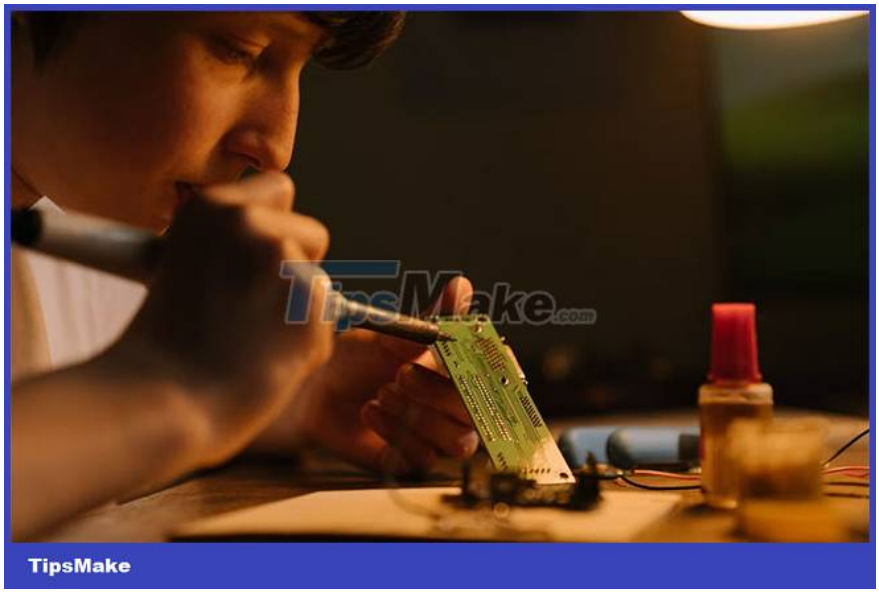
But why would hackers target hardware? The main reason is that hardware is less of a barrier and the device model won't change over the years - for example, there are no hardware upgrades for the Xbox console after release. So an attacker who successfully hacks Xbox 360 hardware could get pretty far before Microsoft releases the next-gen console with better security. Besides game consoles, the same is true for all devices you can think of like laptops, phones, security cameras, smart TVs, routers, and IoT devices.



However, the relative immutability of hardware after production does not mean that they are vulnerable to attack right out of the box. Device manufacturers use components - most notably security chipsets - to ensure their devices remain resilient to almost any attack over a long period of time. Hardware also has firmware (essentially hardware-specific software) that is updated regularly to ensure the device is compatible with the latest software even though its components are old. Firmware updates also make hardware resilient to common hardware hacking methods.

To better understand firmware updates, imagine you have to buy a new game console every time a new type of game comes out. That's not only frustrating, but it's also very expensive. The wiser financial decision would be to buy a console that is compatible with both old and new games or only requires a small fix for full compatibility. For manufacturers, that means they have to anticipate what future generations of games will look like and create consoles that can run them well. Or, at the very least, the components must be compatible with future game releases long enough for the console purchase to be a wise investment.

6 common methods used to hack hardware



Hardware hacking is easy: Hackers need to own, handle, or be in physical range of the device they want to hack. The most common methods used by hackers include opening the device, plugging an external tool into the port, exposing the device to extreme conditions, or using special software. These are common ways attackers hack hardware.

1. Fault Injection

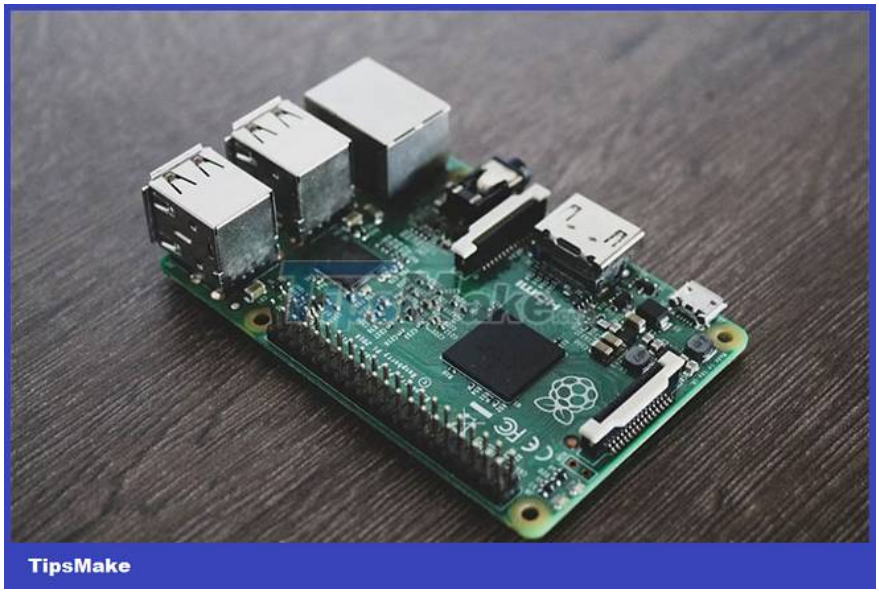
Fault Injection is the act of pressurizing hardware to expose a vulnerability or create a bug that can be exploited. This can be achieved in a number of ways, including overclocking the CPU, repeatedly accessing DRAM, reducing GPU voltage, or causing a short circuit. The goal is to stress the device hard enough to trigger protection mechanisms that don't work as designed. An attacker can then exploit the system reset feature, bypass the protocol, and steal sensitive data.

2. Side-Channel Attack

A Side-Channel Attack is basically exploiting the way the device works. Unlike Fault Injection attacks, the attacker does not have to pressurize the hardware. They just need to observe what makes the system work, how it works, and what exactly happens when the system works or fails.

SCA attacks can take the form of timing program execution, measuring the audio response from failed executions, or measuring the power consumption of a device when performing a specific operation. Attackers can then use these signatures to guess the value or type of data to be processed.

3. Connect to the board or JTAG . port



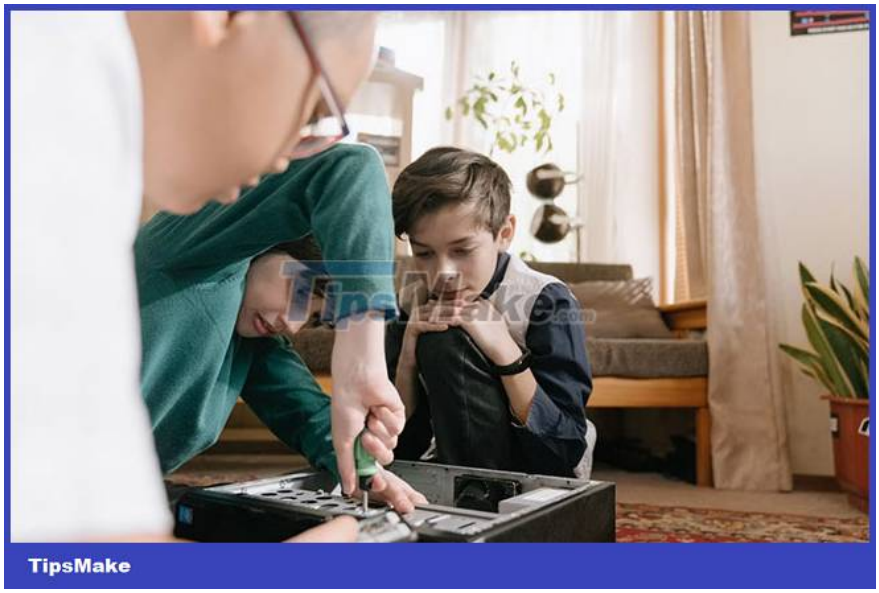
Unlike the aforementioned hardware hacking methods, connecting to the circuit board requires the hacker to open the device. They will then need to study the circuit to find where to connect external modules (like Raspberry Pi) to control or communicate with the target device. A less invasive method is to connect a microcontroller to enable wireless control mechanisms. This particular method works to hack simple IoT devices like coffee makers and pet feeders.

Meanwhile, connecting to the JTAG port takes it up a notch. JTAG, named after its developer, Joint Test Action Group, is a hardware interface on printed circuit boards. This interface is mainly used for low-level programming, debugging or testing of embedded CPUs. By opening the JTAG debug port, hackers can dump (i.e. extract and analyze images of) the firmware for vulnerabilities.

4. Use Logic Analyzer

A logic analyzer is software or hardware for recording and decoding digital signals, although it is mainly used for debugging - like JTAG ports, hackers can use logic analyzers to perform logic attacks. They do this by connecting the analyzer to the debug interface on the target device and reading the data transmitted over the circuit. Usually, this will open the debug console, bootloader, or kernel log. With this access, attackers look for firmware bugs that they can exploit to gain backdoor access to the device.

5. Parts replacement



Most devices are programmed to work specifically with proprietary firmware, physical components, and software. However, sometimes, they also work with generic components. This is a vulnerability that hackers often exploit. Usually this involves replacing the firmware or a physical component - as in the Nintendo Switch modding process.

Of course, device manufacturers hate this and have established anti-tamper measures that make hardware hacking attempts to brick the device. Apple is particularly well known for resisting customers frequently opening or fixing their hardware, even if it's to repair a broken device. You can brick your Apple device if you replace a part with a non-MFI option (made for iPhone, iPad, and iPod). However, anti-tamper measures won't stop hackers from finding vulnerabilities and modifying devices.

6. Taking advantage of the Memory Dump . file

Memory Dumps are files that contain data or log errors that occur when a program or device stops working. Windows computers create dump files when the operating system crashes. The developer can then use these files to investigate what caused the problem in the first place.

But you don't have to be a developer with a lot of knowledge to understand or analyze dump files. There are open source tools that anyone can use to extract and read dump files. For users with certain technical knowledge, the data from the dump file is enough to find problems and solutions. But for a hacker, dump files are a valuable resource to help them discover vulnerabilities. Hackers often use this method to steal Windows login credentials.

Should we be too worried about hardware hacking?

Not really, especially if you're a regular device user. Hacking hardware for malicious purposes presents a high risk to the attacker. Besides leaving a trail that can lead to criminal or civil liability, it is also very expensive: Tools are not cheap, procedures are complicated and time consuming. So unless there are really special interests, an attacker won't target a random person's hardware.

Hardware manufacturers, on the other hand, have to worry about the possibility of such hacks uncovering a trade secret, violating intellectual property rights, or exposing customer data. They need to prevent attacks, push for regular firmware updates, use recoverable components, and set up tamper-proof measures.