5.3 Side-Channel Attack

A side-channel attack is a security exploit that aims to gather information from or influence the program execution of a system by measuring or exploiting indirect effects of the system or its hardware -- rather than targeting the program or its code directly. Most commonly, these attacks aim to exfiltrate sensitive information, including cryptographic keys, by measuring coincidental hardware emissions. A side-channel attack may also be referred to as a *sidebar attack* or an *implementation attack*.

As an illustration, imagine you're trying to determine where a person has driven their car. A typical attack channel would be to follow the car or use a Global Positioning System (GPS) tracker. A side-channel attack, on the other hand, would use measurements about the car to try and determine how it's used. For example, measuring changes in the amount of gas in the tank, car's weight, heat of the engine or passenger compartment, tire wear, paint scratches and the like may reveal information about the use of the car, places or distances it has traveled, or what is stored in the trunk -- all done without directly affecting the car or alerting its owner that they are under investigation.

Historically difficult to do, side-channel attacks are now more common because of several factors. Increasing sensitivity of measuring equipment has made it possible to gather extremely detailed data about a system while it is running. In addition, greater computing power and machine learning enable attackers to better understand the raw data they extract. This deeper understanding of targeted systems enables attackers to better exploit subtle changes in a system.

Attackers can also go after high-value targets, such as secure processors, Trusted Platform Module (TPM) chips and cryptographic keys. Even having only partial information can assist a traditional attack vector, such as a brute-force attack, to have a greater chance of success.

Side-channel attacks can be tricky to defend against. They are difficult to detect in action, often do not leave any trace and may not alter a system while it's running. Side-channel attacks can even prove effective against air-gapped systems that have been physically segregated from other computers or networks. Additionally, they may also be used against virtual machines (VMs) and in cloud computing environments where an attacker and target share the same physical hardware.

Types of side-channel attacks

Bad actors can implement side-channel attacks in several different ways, including the following.

Electromagnetic

An attacker measures the electromagnetic radiation, or radio waves, given off by a target device to reconstruct the internal signals of that device. The earliest side-channel attacks were electromagnetic. van Eck phreaking and the National Security Agency's (NSA) Tempest system could reconstruct the entirety of a computer's screen. Attackers focus modern sidechannel attacks on measuring the cryptographic operations of a system to try and derive secret keys. Software-defined radio (SDR) devices have lowered the barrier of entry for electromagnetic attackers, which can be performed through walls and without any contact with the target device.

Acoustic

The attacker measures the sounds produced by a device. Proof-of-concept (POC) attacks have been performed that can reconstruct a user's keystrokes from an audio recording of the user typing. Hackers can obtain some information by listening to the sounds emitted by electronic components as well.

Power

A hacker measures or influences the power consumption of a device or subsystem. By monitoring the amount and timing of power used by a system or one of its subcomponents, an attacker can infer activity of that system. Some attacks may cut or lower power to cause a system to behave in a way beneficial to the attacker, similar to Plundervolt attacks.

Optical

An attacker uses visual cues to gain information about a system. Although rarely used against computers, some POC attacks have been performed where audio can be reconstructed from a video recording of an object vibrating in relation to sounds. Simple shoulder surfing attacks may also fall into this category.

Timing

A bad actor uses the length of time an operation takes to gain information. The total time can provide data about the state of a system or the type of process it is running. Here, the attacker

can compare the length of time of a known system to the victim system to make accurate predictions.

Memory cache

An attacker abuses memory caching to gain additional access. Modern systems use data caching and pre-fetching to improve performance. An attacker can abuse these systems to access information that should be blocked. The Spectre and Meltdown vulnerabilities that primarily affected Intel processors exploited this channel.

Hardware weaknesses

Hackers can use physical characteristics of a system to induce a behavior, cause a fault or exploit data remanence, which is data that persists after deletion. Row hammering attacks happen when an attacker causes a change in a restricted area of memory by quickly flipping, or hammering, another area of memory located close by on the physical random access memory (RAM) chip. Error correction code (ECC) memory can help prevent this attack. In a cold boot attack, the attacker quickly lowers the temperature of RAM, causing some of the information to be retained after power is removed so the attacker can read it back.