electronic design

print | close

What's the Difference Between Match-on-Host and Match-in-Sensor Fingerprint Authentication?

Electronic Design
Anthony Gioeli
Mon, 2015-11-02 11:51

The biometrics space has recently seen a significant spike in using fingerprint authentication as a simple yet secure method for accessing and protecting data. It's also playing a greater role in safe electronic and mobile transactions.

Using fingerprints for user authentication is immeasurably safer and dramatically easier than requiring users to create, remember, and protect passwords, making it a preferred approach of merchants, banks, users, and third-party clearinghouses. A number of technology advances and aggressive innovation by industry leaders has spawned several different forms of fingerprint recognition, though they're by no means uniform.

Related

What's the Difference Between Secure Comms and Secure Systems?

Sensory's Todd Mozer Discusses Biometric Security For Consumer Devices

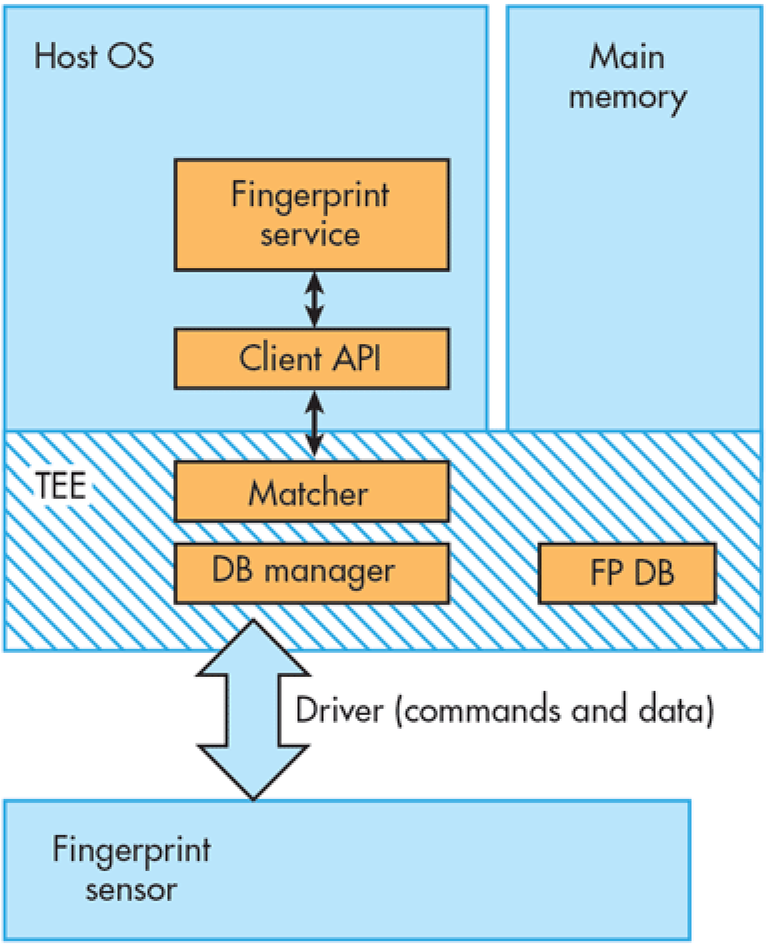
Can The Internet Of Things Be Secure?

One common fingerprint-authentication technique is Match-on-Host technology, whereby the fingerprint sensor reads the fingerprint data and sends it for processing by the host processor or other external processor. While the fingerprint sensor captures the fingerprint data, all of the processing and matching work is performed on the host platform. Then there's Match-in-Sensor technology, a purpose-built, fully encapsulated system-on-a-chip (SoC) architecture that isolates fingerprint enrollment, pattern storage, and biometric matching within the actual fingerprint sensor module.

In the true spirit of "What's the Difference," let's take a look at these two very distinct fingerprint-authentication methods.

Match-on-Host: The Current Standard

The fundamental requirement in fingerprint sensing involves positively identifying the user by making a match with a known and secured "template" or record of the user's fingerprint *(Fig. 1)*. The sensor is used initially to capture the data that creates the user's record in an "enrollment" process, and then gets used during every subsequent access attempt to capture fingerprint data to compare with the stored template.



Virtually every implementation of fingerprint sensing today performs the matching process directly on the host system, whether a smartphone, tablet, PC, or a dedicated device purpose-built for security. As a result, the Match-on-Host architecture splits the functional requirements between the sensor IC that captures the data and a separate controller IC (often the application processor on a mobile device) used to run the software to make the actual match.

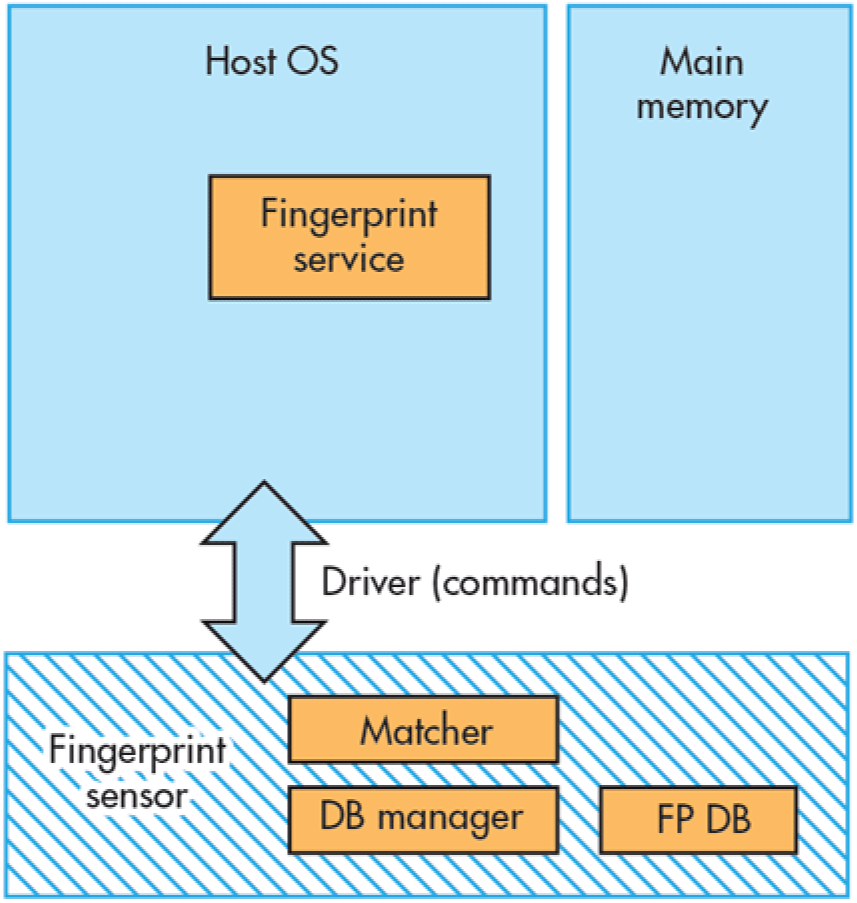
The use of host resources is a natural starting point for any new technology. For this reason, first-generation fingerprint sensors were simple devices limited to a single task: collecting the fingerprint data that would then be used by software running in the host to authenticate the user.

The functions performed in software include identification of fingerprint characteristics, creation of a secure biometric asset (the fingerprint template), storage of the asset, and matching a newly created fingerprint template with the one stored on the device. The host system also provides the security required to protect the integrity and privacy of the fingerprint data. Furthermore, the host system is responsible for detecting biometric forgeries; these so-called anti-spoof techniques are necessary to deter presentation attacks.

Two major selling points of the Match-on-Host architecture have been its low cost and short design-in time, which have enabled fingerprint sensing to be added to devices relatively quickly and cost-efficiently. This momentum, in turn, has facilitated advances made on related fronts, such as the creation of the Universal Authentication Framework (UAF) from the Fast Identity Online (FIDO) Alliance. Yet for all its advantages, when it comes to real security, the Match-on-Host approach pales in comparison to the Match-in-Sensor architecture.

Match-in-Sensor: The Next Generation

the name implies, the Match-in-Sensor architecture integrates the matching and other biometric nagement functions directly into the sensor IC. The IC contains a high-speed microprocessor, storage for instructions and data, secure communications, and high-performance cryptographic capabilities. To achieve this level of integration, while creating a secure execution environment within the sensor IC, Match-in-Sensor employs a SoC architecture *(Fig. 2)*.



Because integrating multiple functions is the raison d’être of integrated circuits, this advance might not seem worthy of being designated a “next-generation” advance. But the level of additional security afforded by fully integrating the sensing and matching functions is significant enough that its expected industry impact should not be understated.

The advanced security with the Match-in-Sensor architecture applies both to the system and to the protection of a user’s unique biometric information. System-level security is enhanced with a range of improvements, including:

- Fingerprint data and execution environment of the fingerprint matcher that are physically isolated from the host’s operating system, affording immunity from hacks or malware on the host.

- The sensor performing biometric identification autonomously, without reliance on input from the host that might be compromised.

- Input parameters for the matcher that are the live fingerprint information, which is captured, encrypted, processed, and protected on the sensor chip and its enrollment templates.

- Ability to accurately verify authenticity, because the identification result is signed using a sensor-specific

ʼvate key derived from the hardware.

- Creation, storage, and management of crypto keys that represent the identity credentials being shared—these keys are also used to sign credentials to prevent malware with false information.

Even if the host is completely compromised by a successful attack of any type or origin, it's extremely difficult to force the matcher to generate a false positive result, replay an old result, or in any other way alter or manipulate the match result. This ensures that an identity-authentication subsystem will remain secure even under a worst-case scenario.

With regard to the user's biometric information, protection is enhanced through a number of features. First, the fingerprint data, including all of the features/characteristics extracted from it and all created templates, is processed *only* within the sensor's on-chip CPU and storage. None of this information is ever shared or exposed to the host device. In addition, the enrollment database is located on private flash memory, isolated and physically accessible only by the sensor. Furthermore, the enrollment templates are encrypted and signed by the sensor using proprietary algorithms and strong cryptographic keys before being stored in the private flash memory.

As with the system-level provisions, even if the host becomes completely compromised by a successful attack, the attacker cannot extract any of user's biometric information, avoiding what would be arguably one of the most invasive forms of identity theft imaginable.

Conclusion

The support for an ecosystem encompassing fingerprint-authentication technology is steadily expanding, with vendors such as Synaptics developing advanced Match-in-Sensor solutions. These solutions, which are FIDO-certified, include the ability to read fingerprints at various angles, options for visual or haptic feedback, and device- or application-specific optimizations. Moreover, Match-in-Sensor doesn't require transporting or sharing biometric information between the fingerprint module and the host device, thereby eliminating the risk of biometric data being stolen if the system is compromised.

So, if you think all fingerprint-authentication techniques are the same, you may want to take a good, long look at the Match-in-Sensor architecture and its capabilities.

Source URL: http://electronicdesign.com/embedded/what-s-difference-between-match-host-and-match-sensor-fingerprint-authentication