insight.tech POWERING IoT AND EMBEDDED INNOVATION

Efficient Security Cuts the Cord for Battery-Powered IoT

For many Industrial IoT devices, batteries or renewable energy are the only practical power source. These may be ad hoc devices like retail beacons deployed quickly and at scale. Remote weather monitoring systems deployed in isolated settings are another example. Or they could be mobile robots that need to disconnect from fixed power to be effective.

Whatever the case, these systems must be able to maximize performance while preserving battery life and remaining secure.

Security is especially challenging in battery-powered systems because of their severe resource constraints. Developers of systems with a fixed power supply tend to focus on issues like size, weight, and heat. But in battery-based designs, total energy consumption is of paramount importance.

Many engineers overlook the fact that simple RSA and ECC cipher suites can consume a considerable amount of energy **(Figure 1)**. Designers of battery-powered systems don't have this luxury.

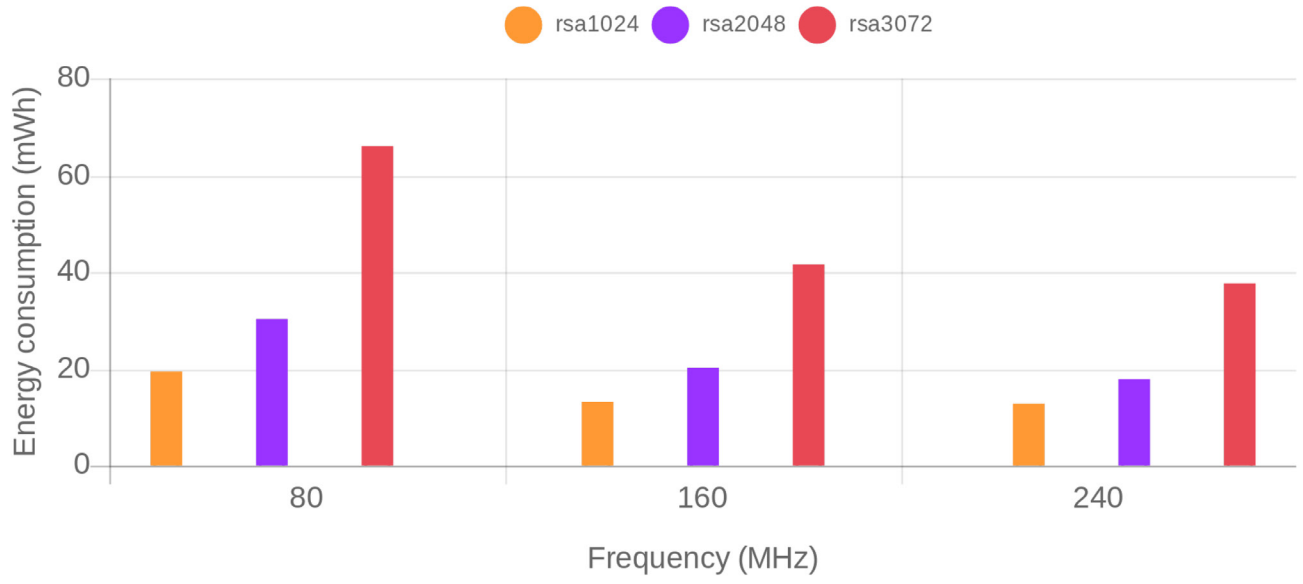

Figure 1. Energy consumption of various cipher suites running on an embedded microcontroller (MCU) at different clock frequencies, represented in milliwatt hours (mWH). (Source: MDPI)

Another consideration for battery-powered IoT systems is the latency of encrypted communications. In general, the longer the encryption key (128-bit, 256-bit, etc.), the stronger the encryption. But there are downsides to longer keys because of the additional time required to encrypt information using them **(Figure 2)**.

Latency can have a waterfall effect on a system, especially if it is battery powered. If the application has real-time requirements, latency constraints may dictate the size of the encryption key used in the design, and therefore the security of the system. Key length also has implications for power consumption, as more clock cycles are needed to encrypt data with longer keys. Memory footprint is another concern of larger keys.

Lowering Security Overhead with a Trusted Platform Module (TPM)

One solution from the PC and networking world offers a possible approach for minimizing security overhead in battery-powered systems. The Trusted Computing Group (TCG) Trusted Platform Module (TPM) specification was formally ratified as an ISO standard in 2009. It specifies a secure co-processor that internally generates and stores cryptographic keys **(Figure 3)**.

In essence, the original TPM standard specified a discrete ASIC that offloads many security operations from the host processor. It also functions as a root of trust (RoT) that can be used to authenticate the integrity of an entire system, including peripherals and accessories.

By moving these functions into a specialized chip, developers can optimize latencies and memory usage without having to move to a larger, more expensive, and more power-hungry host processor.

But a discrete TPM also comes with tradeoffs, such as cost, size (more space is required on the PCB), and its own power consumption. In response, the TCG expanded its offerings to include several different types of TPMs that provide various options in terms of integration, security, and cost.

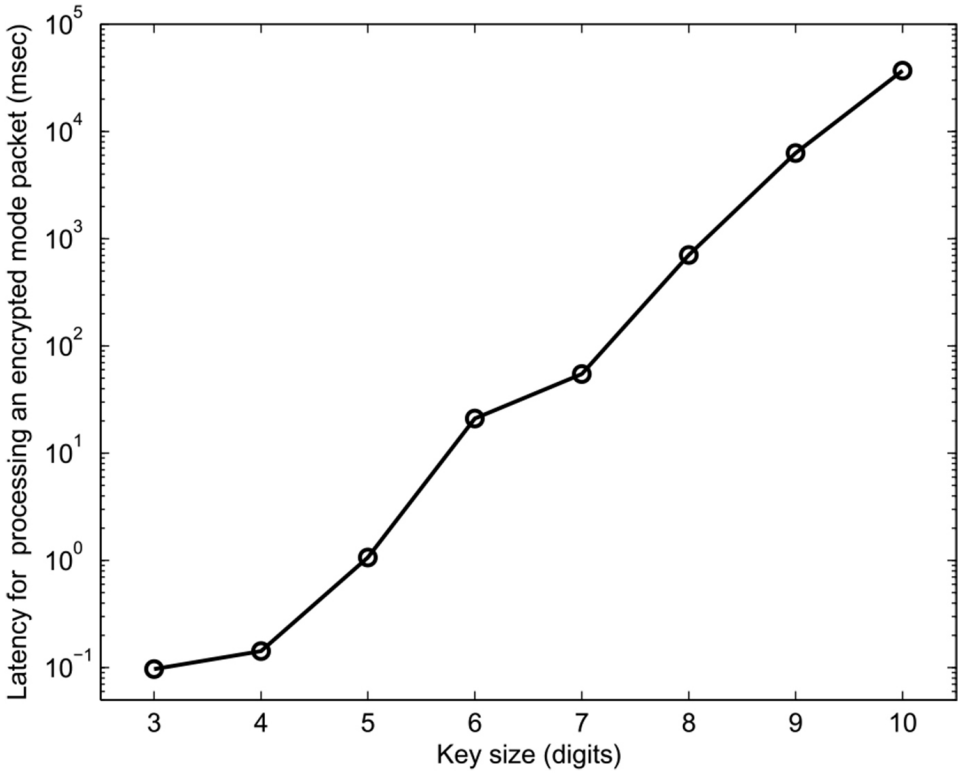

Figure 2. Encryption carries tradeoffs, as longer, more secure keys can add latency, power consumption, and memory requirements. (Source: ScienceDirect)

The most popular of these are outlined in **Figure 4**.

For mobile or battery-powered systems, a firmware TPM makes a lot of sense. It resides in a trusted execution environment (TEE), which isolates it from vulnerabilities in the operating system (OS) and other programs.

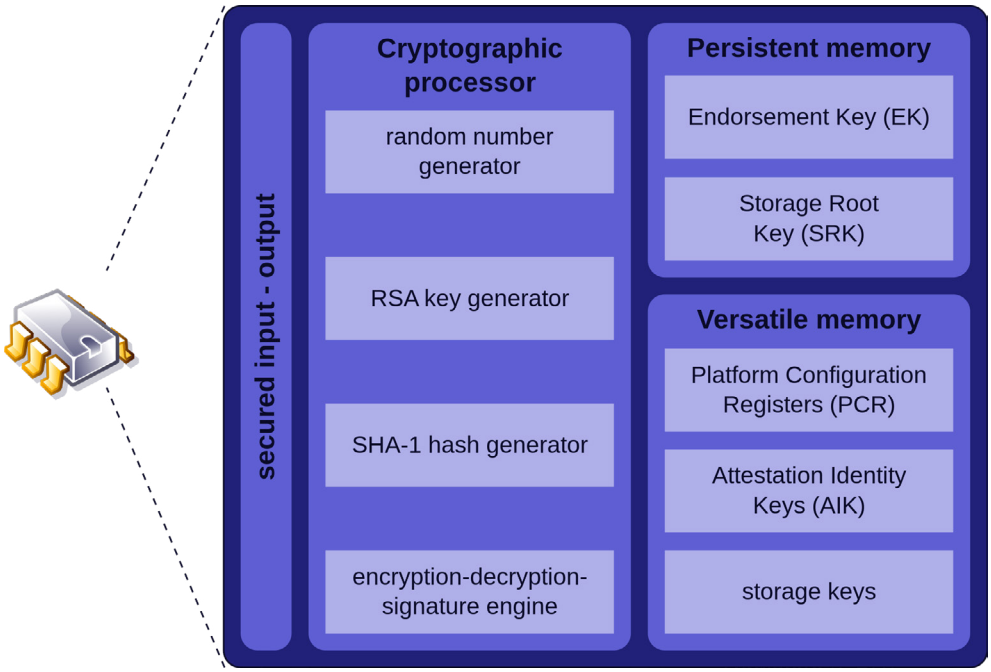

Figure 3. Originally, the Trusted Platform Module (TPM) standard defined a security co-processor that internally generated and stored cryptographic keys. The standard has since evolved to include firmware and software TPM variants as well. (Source: Semantic Scholar)

TRUST ELEMENT	SECURITY LEVEL	SECURITY FEATURES	RELATIVE COST	TYPICAL APPLICATION
DISCRETE TPM	HIGHEST	TAMPER RESISTANT HARDWARE	$$$	CRITICAL SYSTEMS
INTEGRATED TPM	HIGHER	HARDWARE	$$	GATEWAYS
FIRMWARE TPM	HIGH	TEE	$	ENTERTAINMENT SYSTEMS
SOFTWARE TPM	NA	NA	cc	TESTING & PROTOTYPING
VIRTUAL TPM	HIGH	HYPERVISOR	c	CLOUD ENVIRONMENT

Figure 4. The Trusted Computing Group (TCG) Trusted Platform Module (TPM) specification has expanded to include various implementations. (Source: Trusted Computing Group)

Compared to discrete TPMs or TPMs integrated as hardware into a system on chip (SoC), a firmware TPM adds no physical area and reduces power consumption because no additional circuits are required.

While software- and firmware-based encryption has historically incurred massive performance penalties versus hardware solutions (approaching 50 percent in some cases), newer implementations make the disparity negligible (less than 5 percent). In fact, as shown in **Figure 5**, certain implementations of a firmware TPM now actually outperform their discrete TPM counterparts.

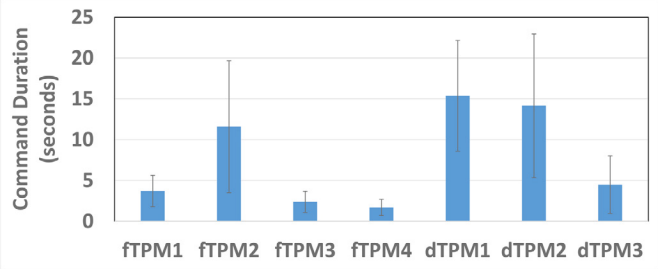

Figure 5. The latency of creating RSA-2048 keys on multiple firmware Trusted Platform Modules (TPMs) and discrete TPMs. (Source: USENIX Association)

This performance improvement also helps offset power consumption in battery-operated devices as the host processor requires fewer clock cycles to execute a TPM command.

Security on Batteries: Flexibility of Firmware, Performance of Discrete TPMs

Intel® Platform Trust Technology (Intel® PTT), based on the TCG's TPM 2.0 specification, is one commercial-grade firmware TPM solution. It is available on many Intel® processors going back three generations, including low-power Intel Atom® SoCs based on the Bay Trail microarchitecture and 8th generation Intel® Core™ U, Y, and M variants.

Intel PTT assumes that root keys will be stored in firmware. Although this level of security isn't the highest possible, it does allow for security patches and updates in the event of an exploit. But clever security architectures also allow designers to store PTT keys in hardware, within a system's memory subsystem.

X1 flash memory controllers from Hyperstone GmbH, for example, essentially provide a hardware assist to firmware TPMs like Intel PTT. For battery-based system designs that require a flash memory controller, the X1 integrates one AES and one ECC module that perform hardware-based encryption and decryption of data stored in memory **(Figure 6)**. This architecture helps with latency but also total energy consumption.

Although the memory controller consumes slightly more power than the host processor during encryption operations, the X1 executes these commands more quickly thanks to the optimized AES and ECC modules. A processor may consume one additional watt per second of encryption over a period of five seconds (5 W total). Even so, the X1 will expend two additional watts but complete the operation in 1.5 seconds (3 W total).

In addition, the described security architecture enables extensible security over time. A fundamental tenet of the TPM 2.0 specification is that it allows for "algorithm agility," or the ability to implement new cryptographic algorithms as needed.

Because the solution is part firmware and part software, it is flexible enough to support the inclusion of new ciphers, algorithms, key sizes, or future changes to PTT.

Hyperstone X1 controllers include APIs for integrating customer or use case-specific security extensions, as well as an ISO 7816 interface for communicating with other security components. The latter helps facilitate certification to standards like the Information Technology Security Evaluation Criteria (ITSEC) or Common Criteria for Information Technology Security up to evaluation assurance level 5 (EAL5).

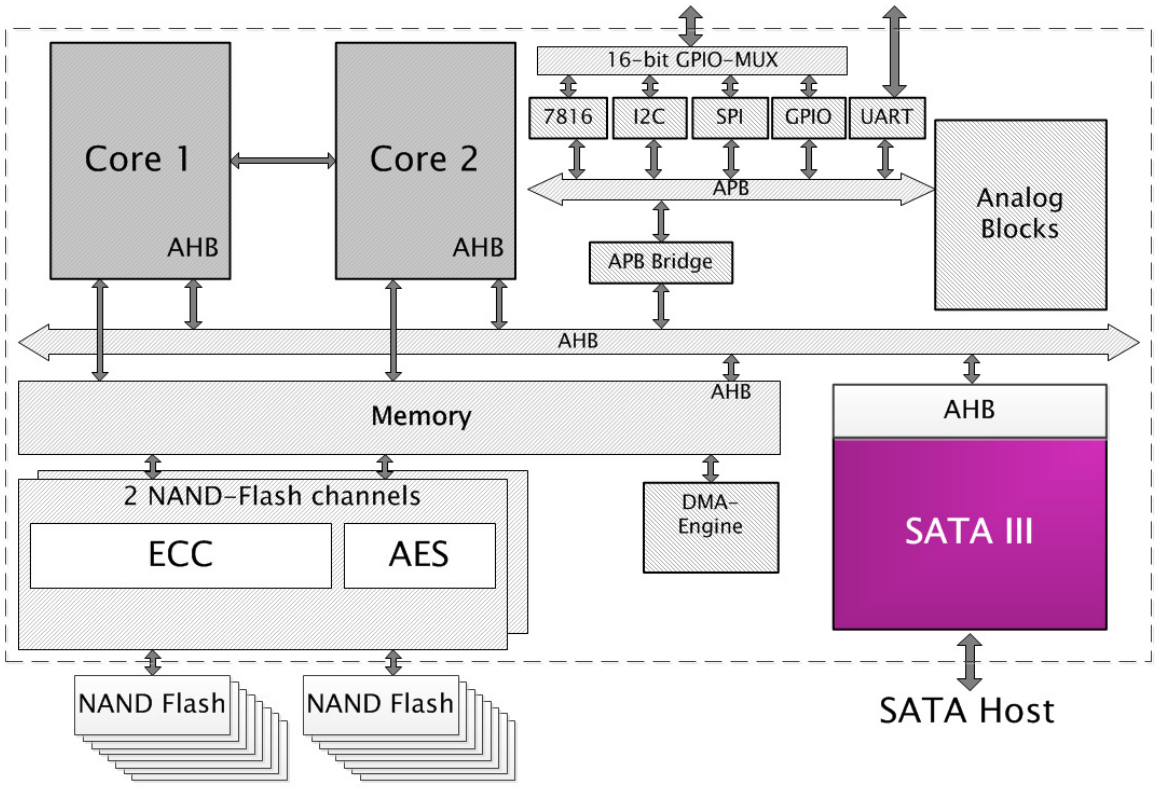

Figure 6. The Hyperstone X1 flash memory controller includes two dedicated hardware encryption blocks, which help reduce latency and energy consumption for Intel® Platform Trust Technology (Intel® PTT) crypto operations. (Source: Hyperstone GmbH)

Energy-Efficient Security Helps IoT Cut the Plug

Many of the PTT-enabled processors mentioned earlier are available on Intel® Next Unit of Computing (Intel® NUC) platforms, which developers in the MakerPro community currently power with 20 V battery packs. Running as a desktop, one user estimated roughly four hours of battery life, which isn't half bad for a hobbyist project with an off-the-shelf bill of materials of a few hundred dollars.

If an organization with commercial aspirations were to use these building blocks as the baseline of a far simpler IoT system, costs would go way down and battery life way up.

With technologies like Intel PTT and the Hyperstone X1 flash memory controller, data on such a system could be captured, stored, and transmitted securely. There would be little concern over power consumption or longevity of integrated encryption technologies. And the solution could pair with platforms like Intel® Cloud Integrity Technology (Intel® CIT) to form the foundations of robust end-to-end system security.

It's time for IoT devices to pull the plug.