Implementation of High-Level Cryptographic Protocols using a SoC Platform

Quentin Delhaye

May 29, 2015

In 1965, Gordon Moore, co-founder of Intel, stated that for the same footprint, the number of transistors in an integrated circuit should double every two years. It became the Moore's law, an empirical prediction that drived inovation for fifty years. Nowadays, it is reaching its limits, as Intel in delaying the manufacturing of its next-generation processors.

The technological limitation is not the only parameter slowing the development of new integrated circuits. The second law of Moore states that the cost to design, manufacture and test the new architectures doubles every four years. This economical growth will eventually collide with the Moore's law, stopping its progression.

The domination of multi-purpose processors is reaching its limits, and even the biggest actors on the market are considering alternatives to the Moore's Law.

A report of the Natural Resources Defense Council forecasts the US datacenter electrity bill to reach $13.7 billion by 2020. As CPUs are largely responsible for the power consumption, it makes their design critical. A solution is to use a hybrid architecture, combining a general-purpose micro-processor with an integrated circuit configurable on-the-fly. This is a path Intel is exploring with its new Xeon product line, a processor aimed at servers, that combines a Xeon CPU with an FPGA from Altera. An FPGA is an integrated circuit that can be reconfigured after the manufacturing. Adding such a device to a micro-processor makes it a true System on a Chip capable to reconfigure itself on-the-fly to accelerate specific operations.

On a different scale, Altera developed an ARM based SoC FPGA platform, combining an FPGA with an ARM Cortex-A9, a low-power processor. This platform allows an increase of performance, lower power consumption and reduced board size compared with a solution with an FPGA separated from the CPU. Such architecture is aimed at embedded and low-energy systems, where not only the performance, but also the power consumption matters.

The aim of this thesis is to implement various popular cryptographic schemes on the board, and make use of the embedded hardware device to improve the performance and decrease the ressources utilization. This work will focus on the network security and the impact of hardware offloading of cryptographic operations.

There are mainly two types of cryptographic schemes: symmetric and asymmetric. The symmetry actually applies to the key used to encrypt the message. In the first case, the same key is used for encryption and decryption and must be kept secret between the two peers. In the second case, there is a pair of keys: one private, only known from its owner, and a public avaible to anyone. When Bob wants to send an encrypted message to Alice, Bob encrypts the message using the public key of Alice. Once encrypted, the message can only be decrypted using the corresponding private key owned by Alice.

Why the need for two paradigms, you ask? Asymmetric cryptography may be practical, but its operations are complex and take time. Symmetric cryptography, however, is much more efficient. The only downside is that a common key must shared between Alice and Bob.

This is where network security protocols come into play. The first step is to establish a connection. To do so, most protocols take advantage of the public key infrastructure associated with asymmetric cryptography to initiate a "key exchange protocol" (*e.g.* Diffie-Hellman) during wich Alice and Bob will agree on a shared secret. From this secret, they derive a key that they will use the encrypt their communication. This work focuses on three protocols: SSH, implemented by OpenSSH, TLS/SSL, implemented by OpenVPN, and IPsec, already implemented in the Linux operating system.

All those encryptions, decryptions, key computations, authentications, and many others can be *offloaded* to a dedicated hardware device, that is, sent to the device for it do the computations instead of the CPU.

This is what the figures show. The figure 1 shows the number of secure connections per minute that can be addressed when using a software or a hardware implementation, for different levels of security. For the highest level, the hardware is able to connect almost six times more clients for nineteen times less CPU usage.

The figure 2 compares different implementations doing a file transfer over different type of secure channels. OpenSSH is interresting if the performance is the priority, but if the efficiency and the power consumption are the main concern, IPsec is a better solution. In any case, OpenVPN is not suited

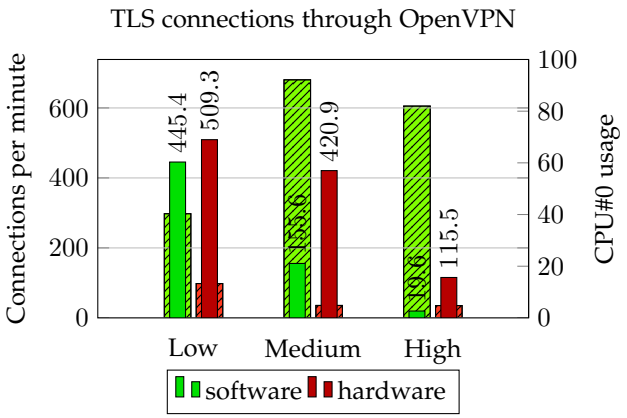

Fig. 1. Comparison of file transfer methods – the background stripped bars are the CPU usage.

for such a platform as it poorly uses the hardware device.

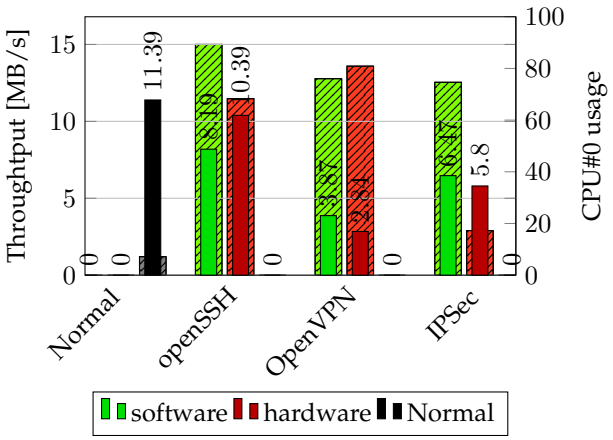

Fig. 2. Comparison of file transfer methods – the background stripped bars are the CPU usage. Low, medium and high correspong to the key size of the assymetric cipher used (RSA).

This thesis is a proof of concept that hardware offloading of cryptographic protocols can add tremendous features to embedded platforms. Be it to improve the performance or decrease the CPU utilization, and thus the consumption of the board, hardware offloading is the way to go if you want your embedded system to become an efficient security platform.