The Power Play: Security-Energy Trade-offs in the IoT Regime

Sandip Ray*, Tamzidul Hoque†, Abhishek Basak‡, and Swarup Bhunia†
* NXP Semiconductors, 6502 William Canon Dr. West, Austin, TX 78735
sandip.ray@nxp.com
‡ Dept. of Electrical Engg. and Computer Sc., Case Western Reserve University, Cleveland, OH 44106
axb594@case.edu
† Dept. of Electrical and Computer Engg., University of Florida, Gainesville, FL 32611
{thoque, swarup}@ece.ufl.edu

***Abstract*—We are in the regime of Internet-of-Things (IoT), — a regime characterized by billions of smart, connected computing devices coordinating to provide large-scale, highly personalized applications. Two overriding themes in this regime are energy consumption and security enforcement, which are both critical to the sustainability and proliferation of the IoT ecosystem. However, energy and security requirements are often at odds. This paper discusses several challenges in developing trustworthy IoT devices that comprehend the energy-security trade-offs. We also outline some emergent approaches to address this conflict.**

I. Introduction

The regime of the Internet-of-Things (IoT) is defined to be the point in time when the number of connected computing devices exceeds the human population [1]. Based on this definition, we are currently deep in this regime (cf. Fig. 1). This is an exciting time for computing, distinguished by an explosive growth in the number of smart, connected computing devices around us. It is estimated that over 50 Billion devices would be deployed and mutually connected by 2020, compared to a "only" 500M devices in 2003. The growth in computing during the IoT era represents the fastest economic growth ever experienced for any sector in the history of human civilization. Furthermore, the scope of computing is also becoming pervasive and diverse in scale. Only a couple of decades earlier, computing devices like phones with a few custom applications represented the boundary of our imagination; in contrast, today we are not only imagining, but actually developing and producing solutions ranging from smart watches, fitness trackers, implants all the way to smart homes, vehicles, multiplexes, highways, and cities.

Energy and security constraints provide some fundamental challenges that must be addressed for successful development and deployment of IoT infrastructures. On the one side, the IoT ecosystem includes billions of smart sensory devices (*e.g.*, thermostats, watches, fitness trackers, etc.) that must perform on a thrifty energy profile while maintaining connectivity and communication with the routers, gateways, and the cloud as well as potentially other devices in the same environment (*e.g.*, other smart devices in the same smart home). A primary reason for the low energy requirement is the need for longer longevity of the energy source such as battery. But there are

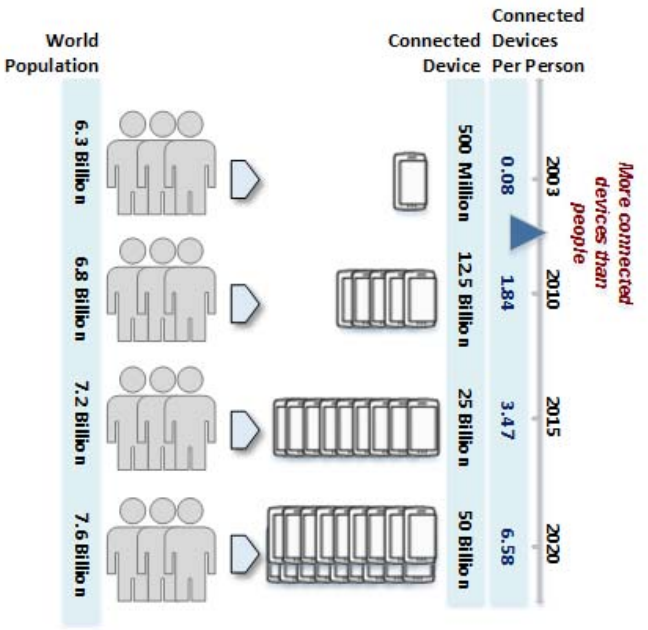


Fig. 1. Growth of connected computing devices vis-a-vis human population over the last decade. Source: Cisco whitepaper [1].

others, *e.g.*, in a wearable device high energy consumption can cause the device to heat up beyond the human tolerance limit, making the device unusable [2]. On the other side, we are for the first time encountering the reality of hundreds to thousands devices computing continuously, monitoring our sensitive, personalized, and private information (*e.g.*, sleep pattern, health, finances, location, browsing preferences, etc.) while being connected to and communicating with billions of other, potentially malicious devices and the cloud. A security compromise requires only a few vulnerable agents in this gigantic communication infrastructure, and the consequence of a security violation can range from compromise of personal end-user privacy to a breakdown of a whole city or multiplex.

There has been significant research on both security and energy constraints in the IoT community. In security, there

978-1-5090-5142-7/16/$31.00 ©2016 IEEE

has been research in robust cryptographic implementations, design obfuscation technologies, security architectures for enforcing system-level access requirements, etc. In energy, there have been advances in low-energy implementations of different functional features, smart clock-gating and power optimization, opportunistic energy harvesting, etc. However, in current research, these two disciplines are quite isolated and the techniques developed for one rarely take into account the constraints in the other. Indeed, the trade-offs between energy and security are not widely understood or appreciated.

Why do we have trade-offs between energy and security constraints? In order to optimize for energy consumption, one often designs a hardware IP[1] to optimize power in frequently occurring cases; this in turn enables for example, a side channel adversary to identify the internal execution information by profiling power consumption. Indeed, much of the current research in security has focused on balancing out energy consumption profiles for different use cases in order to prevent such side channel attacks [3]. Note, however, that the obvious approach of securing designs by eliminating or reducing energy optimization for frequent cases would not work; while successfully addressing the above side-channel attack, such a step might make the design impracticable for deployment by simply pushing the energy profile to be too high for its usage requirements. On the flip side, security architectures have traditionally been developed with heavy-weight trusted execution environments [4], [5], [6] which cannot be easily migrated to a design with aggressive energy constraints. Consequently, developing energy-aware security architecture and security-aware energy optimization is clearly a fundamental requirement for the design and deployment of IoT systems in practice.

Unfortunately, the challenge of security-energy trade-off involves significantly higher complexity than perhaps indicated by the example above. One crucial requirement here is configurability: the energy and security requirements may also need to be adapted on-field or even during a single execution. Consequently, any architecture putatively solving energy-security trade-off issues must additionally be adaptable to changing needs on both sides *post-deployment*.

In this paper we discuss the trade-off challenges between energy and security constraints in IoT systems. Our focus in particular is on dynamic, run-time configurability of security and energy requirements. We discuss a low-cost security solution, which, when implemented in a reconfigurable hardware such as FPGA, can provide a potential solution to the security-energy trade-offs. We discuss some preliminary investigations in this area.

The remainder of the paper is organized as follows. Section II provides a basic, general overview of the IoT, focusing in particular on energy and security constraints. Section III motivates the need for a configurable architecture for enforcing energy-security trade-offs. Section IV discusses a flexible security architecture called E-IIPS, and explains how such an architecture can be used to address the configurability need. We conclude in Section V.

[1]An IP or "Intellectual Property" is a hardware or software component designed for a specific function. A typical SoC design is developed by integration of such pre-designed IPs.

II. IoT Basics

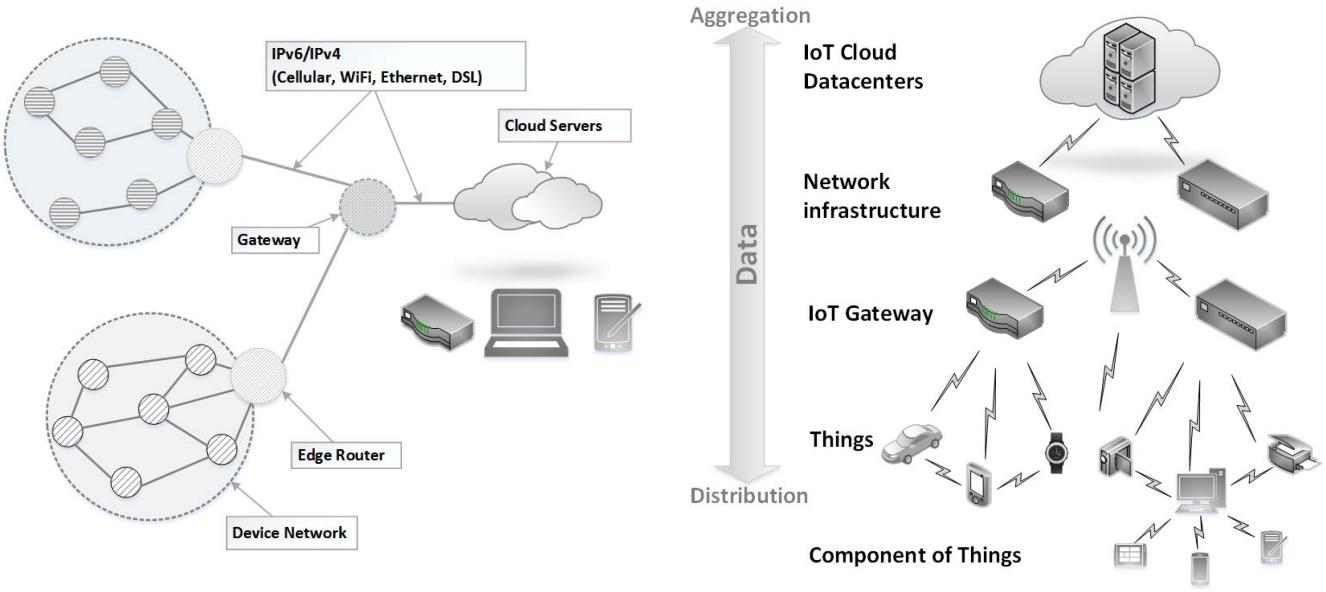
We start with a brief, high-level overview of IoT, focusing primarily on aspects relevant to security and energy constraints. The reader interested in a broader overview of IoT challenges is referred to a recent tutorial [7] for a more comprehensive treatment. Fig. 2 shows the basic elements of the IoT ecosystem, and arranges them into a hierarchy based on data aggregation. In a simplistic view, an IoT application includes devices or "things" with connected sensors that accumulate sensory information from physical world (*e.g.*, location, motion, temperature, etc.) which are aggregated and consolidated through routers, gateways, and the network and communicated to the cloud or datacenters for performing data analytics; the result of the analytics is typically a response, adapting the devices better to the environment or providing feedback to the user of the application (*e.g.*, adjusting the temperature in the thermostat, detecting intrusion in a home surveillance system, etc.).

Energy thriftiness in the IoT are crucial along the "things" and sensors, although with increasing adoption the criticality is extending to the network and cloud as well. Due to the mobile and sometimes standalone nature of these devices, powering them poses a new paradigm in power delivery and management solutions. The increasing demand for features and intelligence in IoT devices further exacerbates the problem. Fine-grained spatio-temporal power gating and clock gating are already in widespread use in the industry. However, we are now faced with decreasing die-sizes, lower decoupling capacitance, multiple chip and platform power states, increasing number of power grids, and migrating hotspots. Consequently, delivering power efficiently is a critical design challenge. In addition, much of the Internet that we know today is not optimized for low-power devices: the default assumption is that devices are always active. For example, TCP cannot distinguish between packets dropped due to congestion or packets lost on wireless links.

Security in the IoT ecosystem, on the other hand, arises from the reality of a large number of connected devices. With so many connected devices communicating from all around the globe, one expects the communication infrastructure to include some malicious components at all times. The problem is acute since the data being communicated is often sensitive, ranging from highly personalized consumer information such as health and sleep pattern, to trade secrets of the enterprise to state secrets of the government and the military. Furthermore, since the devices themselves are complex, architecting their security mechanisms to ensure protection of all assets while still ensuring interoperability and adequate functionality of the device is a challenge.

III. Need for Configurable Security-Energy Trade-offs

The energy and security challenges discussed in the preceding section are crucial to IoT systems, but not germane to



(a) (b)

Fig. 2. (a) The structure of IoT. (b) Data aggregation along different IoT components.

them. Such issues could exist in any low-power, embedded computing device containing sensitive security assets. However, one critical factor distinguishes IoT challenges from traditional embedded devices, *viz.*, the need for configurability after deployment and during execution. In this section, we motivate this need. We will discuss a potential approach to address the energy-security trade-offs under configurability needs in the subsequent sections.

The need for configurability i.e. on-field adaptation of security and energy constraints primarily arises from the changing and evolving needs over the life-time of the device. In particular, note that an IoT application such as a car, home, or multiplex has a long life-time, *e.g.*, two or more decades. Compare this to our smartphones and tablets, which have a life-span of no more than a couple of years. The problem with applications having a long life-span is that user needs change over this period, *e.g.*, security or energy needs in computing today are very different from those a decade back. In addition, the IoT regime for the first time involves communication devices that were never designed with the intention to be connected to the Internet, e.g., light bulbs, refrigerator, car, etc. It is impossible to comprehend either the different usage models or the energy requirements and exploitation scenarios involved in such a connection. For example, we do not know the potential usages of a refrigerator communicating with a (self-driving) automobile regarding restocking, nor the repercussions of a compromised light-bulb eavesdropping on such a communication. Consequently, even the energy and security specifications and requirements, let alone their implementations, must remain fluid at design or deployment time and our critical recourse is to develop highly flexible on-field patching and update mechanism when a new usage or new exploitation is discovered.

In addition to the above, energy and security requirements may need to adapt *dynamically*, even within a single execution due to changing environment. Consider a smart watch being used by a mobile user. Depending on the location, the energy and security constraints may be different, *e.g.*, when using it outdoors through a public network the user may require stronger (and potentially more energy-consuming) authentication than when the same watch is used at home or work via a more trusted network. Note that "less energy inside and more energy outside" is not a universal rule that can be designed into the system. For example, consider two users, one who rarely uses the Internet outdoors (and if so, for no communication involving any sensitive assets) and another who uses the watch as a fitness tracker continually synchronizing the captured data to the cloud for providing effective feedback regarding health and fitness. For the former user, energy constraints rather than security may be more essential during outdoor activity, while for the latter the priorities may depend on the degree of protection the user wants for personal fitness and activity data vis-a-vis extended battery life for the watch. The optimal strategy for each user is probably unique, and likely determined by learning from the user's preference over time.

IV. A Configurable Security Architecture

The security-energy trade-offs above suggest the need for an architecture to implement those constraints in a way that permits adaptation and configuration post-silicon and on-field while being aware of the overall limitations of energy consumptions imposed by the device. In this section, we discuss one approach we are investigating to address this question.

Our response to this requirement is to develop a centralized hardware IP for enforcing security and energy (among other

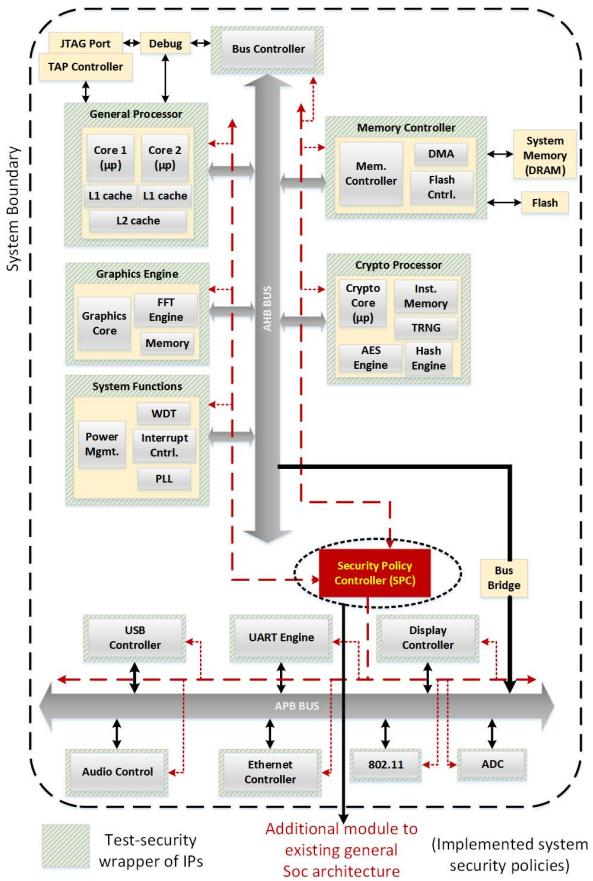


Fig. 3. An overview of the E-IIPS architecture that enables such trade-off [8].

non-functional) constraints. This approach is derived from our previous work on developing a flexible, configurable architecture for security policy implementations [8], [9], [10]. Security policies are rules or constraints that specify how different sensitive assets in the SoC design can be accessed at different points of the execution or different stages of the system life-cycle. Our architecture, called E-IIPS or extended infrastructure IP for security, includes (1) a centralized policy control unit called "Security Policy Engine" (SPE for short), and (2) a collection of wrappers for different IPs that detect relevant security-critical events. Fig. 3 provides a high-level overview of the architecture. The SPE is implemented as a microcontroller, which can be programmed by the security architect with diverse security requirements.

How does the E-IIPS architecture enable security-energy trade-offs? First, constraints relevant to power-management can also be programmed in SPE, as long as the environmental conditions triggering these events can be effectively detected. Second, since the implemented constraints are programmable through the microcontroller, they can be updated in field. Third, programmability permits adaptation of the energy-security trade-offs during execution, and even facilitates learning from historical data based on user's preferences.

On the other hand, in addition to inheriting traditional software/firmware vulnerabilities, micro-controlled designs typically incur high energy cost making them unsuitable for IoT devices. It is therefore critical for the design to provide low-energy on-field patching capabilities while ensuring adherence to strict energy budget. Our initial experiment suggests that while the energy cost is not excessive, it is substantial and may be prohibitive for devices with aggressive energy constraints.

We are addressing this problem by re-designing E-IIPS to be implemented on a reconfigurable hardware block such as an embedded domain-specific FPGA. FPGA's also provide programmability and on-field patching infrastructure, albeit more complex than that of a microcontroller. On the other hand, FPGA implementations can be significantly energy-efficient, enabling potential implementation of E-IIPS architecture on highly energy-thrifty IoT devices while still maintaining on-field programmability and adaptation.

V. Conclusion

We have described some of the challenges in the trade-offs between energy and security constraints in the IoT regime. We discussed the critical role of configurability and adaptability of any architecture developed to enforce these trade-offs and presented an attempt at developing such an architecture. Our approach to develop a centralized policy controller based on a reconfigurable hardware fabric (e.g. FPGA) shows promise to address the key trade-offs in the IoT era. In spite of the promise, it must be admitted that E-IIPS — in particular its FPGA implementation — is very much a work in progress. A comprehensive experimentation with the architecture is necessary to ensure that the approach suits the diverse constraints of energy-security trade-offs.

VI. Acknowledgement

The work is supported in part by US National Science Foundation (NSF) Grants 1603475 and 1603483, and Semiconductor Research Corporation grants 2015-EP-2650.

References

[1] D. Evans, "The internet of things - how the next evolution of the internet is changing everything," *White Paper. Cisco Internet Business Solutions Group (IBSG)*, 2011.
[2] "Basis Tells Customers to Stop Wearing Its Peak Watch," 2016, http://www.consumerreports.org/fitness-trackers/basis-tells-customers-to-stop-wearing-peak-watch.
[3] A. Rane, C. Lin, and M. Tiwari, "Raccoon: Closing Digital Side-Channels through Obfuscated Execution," in *24th USENIX Security Symposium*, J. Jung and T. Holz, Eds., 2015, pp. 431–446.
[4] Intel, "Intel® Trusted Platform Module Hardware User's Guide," http://download.intel.com/support/motherboards/server/sb/g21682003_tpm_hwug.pdf.
[5] ——, "Intel® Software Guard Extensions Programming Reference," https://software.intel.com/sites/default/files/managed/48/88/329298-002.pdf.
[6] Samsung, "Samsung KNOX," www.samsungknox.com.
[7] S. Ray, Y. Jin, and A. Raychowdhury, "The Changing Computing Paradigm with Internet-of-Things: A Tutorial Introduction," *IEEE Design & Test of Computers*, vol. 33, no. 2, pp. 76–96, 2016.
[8] A. Basak, S. Bhunia, and S. Ray, "A Flexible Architecture for Systematic Implementation of SoC Security Policies," in *ICCAD*, 2015.
[9] S. Ray, J. Yang, A. Basak, and S. Bhunia, "Correctness and Security at Odds: Post-silicon Validation of Modern SoC Designs," in *Design Automation Conference*, 2015.
[10] A. Basak, S. Bhunia, and S. Ray, "Exploiting design-for-debug for flexible SoC security architecture," in *DAC*, 2016, pp. 167:1–167:6.