Research Activities on FPGA Design, Cryptographic Hardware, and Security Services

Alessandro Cilardo
Department of Computer and System Engineering, University of Naples Federico II
via Claudio 21, 80125 Napoli, Italy, Email: acilardo@unina.it

***Abstract*—This paper reports on the main research results achieved by the author, including activities carried out in the context of funded Research Projects, until year 2012. The report presents an overview of the findings involving cryptographic hardware, as well as the results related to the acceleration of cryptanalytical algorithms. Another major research line involved FPGA design automation and testing. The above results were complemented by works on security service provisioning in distributed environments. The report presents an exhaustive description of all the scientific works derived from the above activities, indicating the essential insights behind each of them and the main results collected from the experimental evaluation.**

I. Overview

This paper provides a report of the main research results achieved by the author, including activities carried out in the context of funded Research Projects, until year 2012. A major line of research involved the design of advanced hardware blocks for cryptographic applications, motivated by the fact that hardware devices provide both high performance and resistance to tampering attacks, and are thus ideally suited for implementing computationally intensive cryptographic routines handling sensitive data. Most of the works developed by the author deal with reconfigurable hardware devices, i.e. Field Programmable Gate Arrays (FPGAs), although a few results targeted Application Specific Integrated Circuits (ASICs). The research activity also covered the use of dedicated hardware solutions for cryptanalysis, the set of techniques aimed at breaking cryptographic algorithms in order to demonstrate possible weaknesses. Today's acceleration technologies can in fact play a key role for cryptanalytic applications. In this context, the author's activity aimed at exploring the adoption of advanced compute platforms, based either on software-programmable or hardware-reconfigurable acceleration solutions, for cryptanalytical purposes. As a further research line, the activity investigated new techniques for FPGA design automation and testing. The most significant results included methodologies for early estimation of hardware complexity in high-level synthesis processes, as well as a range of techniques specifically targeted at the test of hardware-reconfigurable devices. Last, a different part of the research activities dealt with security services, particularly public-key infrastructures and digital time stamping, provisioned to heterogeneous mobile devices. Delivering security services to such classes of devices, in fact, raises a number of challenging issues, mostly related to the limited amount of computing power typically available on those platforms. Hence, in addition to hardware-related topics, the author's activity also covered the implementation, deployment, and evaluation of security services.

Manuscript revised December 31, 2012.

This paper is organized as follows. Section II presents an overview of the main results involving cryptographic hardware. Section III summarizes the results related to the acceleration of cryptanalytical algorithms. Section IV reports on the activity involving FPGA design automation and testing. Section V describes the essential insights behind the works on security service provisioning in distributed environments. Section VI concludes the paper with a few final remarks.

II. Cryptographic hardware

A major line of research involved the design of advanced hardware blocks for cryptographic applications. In fact, hardware devices provide both high performance and resistance to tampering attacks, and are thus ideally suited for implementing computationally intensive cryptographic routines which operates on sensitive data.

In [1] the author presented a hardware implementation of the Rivest-Shamir-Adleman (RSA) algorithm for public-key cryptography. Basically, the RSA algorithm involves a modular exponentiation operation on large integers, which is considerably time-consuming to implement. To this end, we adopted a novel algorithm combining the Montgomery's technique and the carry-save representation of numbers. A highly modular, bit-sliced architecture was designed for executing the algorithm in hardware. We also proposed an FPGA-based implementation of the architecture developed. The characteristics of the algorithm, the regularity of the architecture, and the data-flow aware placement of the FPGA resources resulted in a considerable performance improvement, as compared to other implementations presented in the literature.

Differently from the previous work, in [2] the author presented a hardware architecture and an FPGA-based implementation of the Montgomery's algorithm relying on a *digit-serial* approach, which allows the basic arithmetic operations to be broken into words and processed in a serialized fashion. As a consequence, the architecture implementation takes advantage of short critical paths and low area requirements. In fact, as compared to other solutions in the literature, the proposed implementation of the RSA processor achieved smaller area requirements and comparable performance. We thoroughly

explored the design trade-offs, in terms of area cost vs. time performance, for different values of the key length and the serialization factor of the serial architecture, and the final performance level was given as a function of this factor.

Generalizing on the previous results, the activity targeted the systematic exploration of the design space for FPGA-based implementation of RSA. The journal paper [3] presented two alternative architectures for implementing the RSA algorithm on reconfigurable hardware. The two solutions were at the extremes of the design space, since one adopted a word serial approach, while the other had a fully parallel organization. Based on the analysis of these architectures for different values of the serialization factor, we explored the design space for the FPGA-based implementation of the RSA algorithm. We analyzed and compared the results of the two design processes with respect to two fundamental metrics, execution time and FPGA resource usage. We emphasized pros and cons and commented on the trade-offs of the two design alternatives.

After focusing on RSA, the activity moved to other cryptosystems, particularly Elliptic Curve Cryptography (ECC). In fact, ECC has gained widespread exposure and acceptance, and has been included in many security standards. In [4] the author reviewed the essential insights behind ECC implementation, as a prominent case study of *cryptographic engineering*, a complex, interdisciplinary research field encompassing such areas as mathematics, computer science, and electrical engineering. In particular, the work showed that the requirements of efficiency and security considered at the implementation stage affect not only mere low-level, technological aspects but also, significantly, higher level choices, ranging from finite field arithmetic up to curve mathematics and protocols.

Interestingly, ECC enlarges the spectrum of the underlying mathematical operations to be supported. For their performance, Elliptic Curve cryptosystems are critically dependent on modular multiplication, performed in one of two different algebraic structures, $GF(N)$ and $GF(2^m)$, which normally require distinct hardware solutions for speeding up performance. For both fields, Montgomery multiplication is the most widely adopted solution, as it enables efficient hardware implementations. In [5] the author presented a novel unified architecture for public-key cryptography. Based on a fully-parallel, bit-sliced unified scheme, the architecture was designed to perform integer modular multiplication/exponentiation used in $GF(N)$ as well as $GF(2^m)$ multiplication, the core operations of RSA and EC cryptography. The architecture used a radix-2 Montgomery technique for modular arithmetic, and a radix-4 most significant digit (MSD)-first approach for $GF(2^m)$ multiplication. The bit-sliced scheme was highly regular, modular, and scalable, as virtually any datapath length could be obtained at a linear cost in terms of hardware resources and no costs in terms of critical path. The proposed solution outperformed all similar unified architectures found at the time in the technical literature in terms of clock count and critical path. The architecture was implemented on an FPGA device. A highly compact and efficient design was obtained taking advantage of the architectural characteristics.

A further development along this direction was achieved in [6], proposing a novel unified architecture for parallel Montgomery multiplication supporting both $GF(N)$ and $GF(2^m)$ finite field operations. The hardware scheme interleaved multiplication and modulo reduction. Furthermore, it relied on a modified Booth recoding scheme for the multiplicand and a radix-4 scheme for the modulus, enabling reduced time delays even for moderately large operand widths. In addition, the work presented a pipelined architecture based on the parallel blocks previously introduced, enabling very low clock counts and high throughput levels for long operands used in cryptographic applications. Experimental results, based on 0.18 μm CMOS technology, proved the effectiveness of the proposed techniques, and outperformed the best results previously presented in the technical literature.

The activity also focused on low-level arithmetic operations, particularly integer addition on long operands, which is particularly recurrent in cryptographic applications. In [7] the author presented a new speculative addition architecture suitable for two's complement operations. The speculative approach allows shorter combinatorial propagation delays and hence faster circuits, although these circuits might occasionally generate wrong results that need to be corrected. Existing architectures for speculative addition were all based on the assumption that operands have uniformly distributed bits, which rarely occurs in real applications. As a consequence, they were disadvantageous for real-world workloads, although in principle faster than standard adders. To address this limitation, the work introduced a new architecture based on an innovative technique for speculative global carry evaluation. The proposed architecture solved the main drawback of previous schemes and, evaluated on real-world benchmarks, it exhibited interesting performance improvements compared to both standard adders and alternative architectures for speculative addition.

The activity also had a theoretical development. In [8], the author of this report introduced a change of representation for elements in $GF(2^m)$. The proposed representation is useful for architectures that implement unified Montgomery multiplication in finite fields $GF(2^m)$ and $GF(N)$ used for elliptic curve cryptography since it transforms a standard $GF(2^m)$ multiplication into a Montgomery multiplication and comes at virtually no cost in terms of conversion operations.

While focused on implementation aspects, the above activities allowed a deeper understanding of the mathematical basic building blocks in integer modular arithmetic and $GF(2^m)$ arithmetic. This enabled the development of robust results based on a formal approach, leading to two different Transactions journal papers. In particular, in [9] the author presented an efficient bit-parallel $GF(2^m)$ multiplier for a large class of irreducible pentanomials used to build the polynomial representation of the finite field, relying on the newly introduced Shifted Polynomial Bases (SPBs). The theoretical part of the work derived a closed expression of the reduced SPB product for a class of polynomials $x^m+x^{k_s}+x^{k_{s-1}}+\ldots+x^{k_1}+1$, with $k_s-k_1 \leq (m+1)/2$, and then applied the formulation to the case of pentanomials. The resulting multiplier outperformed, or was as efficient as the best proposals in the technical literature, but it was suitable for a much larger class of pentanomials than those studied previously. This property enabled the choice of pentanomials optimizing different field

operations (for example, inversion), yet preserving an optimal implementation of field multiplication, as discussed and quantitatively proved in the last part of the paper.

The second Transactions paper explored an unconventional computational model for the implementation of cryptographic primitives [10]. In fact, motivated by the emerging interest in new VLSI processes and technologies, such as Resonant Tunneling Diodes (RTDs), Single-Electron Tunneling (SET), Quantum Cellular Automata (QCA), and Tunneling Phase Logic (TPL), the author investigated the application of the non-Boolean computational paradigms enabled by such new technologies. In particular, we considered Threshold Logic functions, directly implementable as primitive gates in the above-mentioned technologies, and studied their application to the domain of cryptographic computing. From a theoretical perspective, the work presented a study of the computational power of linear threshold functions related to modular reduction and multiplication. We established an optimal bound to the delay of a threshold logic circuit implementing Montgomery modular reduction and multiplication. In particular, we showed that fixed-modulus Montgomery reduction can be implemented as a polynomial-size depth-2 threshold circuit, while Montgomery multiplication can be implemented as a depth-3 circuit. The work also proposed an architecture for Montgomery modular reduction and multiplication, which ensures feasible $O(n^2)$ area requirements, preserving the properties of constant latency and a low architectural critical path independent of the input size n. We compared this result with existing polynomial-size solutions based on the Boolean computational model, showing that the presented approach had intrinsically better architectural delay and latency, both $O(1)$.

III. Acceleration of Cryptanalysis algorithms

Modern acceleration and heterogeneous computing technologies can have a number of inherent advantages for *cryptanalytic* applications, aimed at breaking cryptographic algorithms in order to demonstrate possible weaknesses. The author's activity was also aimed at exploring the adoption of advanced compute platforms, based either on software-programmable or reconfigurable hardware solutions, for cryptanalytical purposes.

The work in [11] developed a cellBE-based HPC application aimed to gain a deeper understanding of the robustness and weaknesses of the SHA-1 cryptographic hash function. In fact, in the light of previous attacks to the MD5 hash function, SHA-1 remained at the time the only function that could be used in practice, since it was the only alternative to MD5 in many security standards. The work presented a study of the vulnerabilities in the SHA family, namely the SHA-0 and SHA-1 hash functions, based on a high-performance computing application run on the MariCel cluster available at the Barcelona Supercomputing Center. The effectiveness of the different optimizations and search strategies that were used was validated by a comprehensive set of quantitative evaluations. Most importantly, at the conclusion of our study, we were able to identify an actual collision for a 71-round version of SHA-1, the first ever at the time of writing.

Subsequently, the work in [12] moved to the exploitation of reconfigurable hardware for high-performance cryptanalysis of SHA-1. The work explored this opportunity by developing new approaches inherently based on hardware reconfigurability, enabling algorithm and architecture exploration, input-dependent system specialization, and low-level optimizations based on static/dynamic reconfiguration. As a result, the author identified a number of new techniques, at both the algorithmic and architectural level, to effectively improve the attacks against SHA-1. The work also defined the architecture of a high-performance FPGA-based cluster, achieving the highest speed/cost ratio for SHA-1 collision search available at the time. A small-scale prototype of the cluster enabled us to reach a real collision for a 72-round version of the hash function.

IV. FPGA design automation and testing

As a further research line, the activity investigated new methodologies and techniques for FPGA design automation and testing. The work in [13] addressed high-level synthesis, allowing high-level language (HLL) programs to be automatically translated to hardware description language (HDL) modules. The work particularly focused on the problem of estimating as soon as possible the hardware cost resulting from the translation process provided by existing tools. Such *early prediction* of hardware complexity is essential in driving hardware/software partitioning choices, where only a subset of the high-level program is implemented as a dedicated circuit. In that respect, early prediction helps estimate the hardware cost of a given high-level code segment before the (expensive) low-level logic synthesis, dramatically reducing the time required for an exhaustive exploration of different design choices. Clearly, this early estimation is inherently influenced by the specific toolchain for HLL-to-HDL translation. As a consequence, suitable early prediction metrics should be studied and carefully selected for each given toolchain. In the above paper, the author proposed a general framework for the systematic study of such metrics. Unlike some previous works, the proposed framework was not specific to a given toolchain as it enabled designers to plug their own synthesis tool and characterize its behavior in order to identify the most effective metrics to be used during the design space exploration. The framework was developed on top of the LLVM compiler infrastructure along with the R statistical package used to perform regression analysis. For a specific HLL-to-HDL compiler chosen for tests, we collected extensive experimental results on a large base of benchmarks, which showed interesting accuracy improvements over some related work previously presented, confirming the effectiveness of the framework in deriving a characterization of the underlying hardware compiler.

In addition to techniques for FPGA design, the activity also explored new implications of hardware reconfigurability for testing. The work in [14], developed in the framework of the PRIN project *COMMUTA: Hardware/software mutant components for distributed, dynamically reconfigurable systems*, made an attempt to blend together the concept of mobile agents and hardware reconfigurable systems to achieve self-healing

properties. The mobile agent paradigm can potentially handle the complexity and heterogeneity of networked infrastructures, while runtime reconfigurable systems can provide flexible, adaptable, and high performance features. The paper presented a broad analysis of how the innovative aspects of these technologies could be exploited to implement effective test and repair strategies.

Subsequent works addressed specific testing techniques enabled by hardware reconfigurability. In fact, FPGA testing poses a number of challenges related to both the complexity of the device under test and the opportunities introduced by the support to reconfiguration. While techniques for offline testing of FPGAs, either manufacturing-oriented or application-oriented, are relatively mature, in critical applications such as avionics, space, and even numerous commercial products it is often necessary to perform *online* testing. The work in [15] presented a technique for online testing of digital designs implemented on an FPGA. The approach enabled application-oriented testing, in that it covered the subset of the FPGA which is actually used for the implemented design, and considered scenarios where the FPGA component is a part of a larger embedded system. The proposed approach was in fact based on a software framework, acting as an abstraction layer for reconfigurable hardware resources. Essentially, the framework exposed to software applications a Register-Transfer Level view of the underlying hardware, allowing test procedures to be implemented as software programs. The approach proved to be especially advantageous when memory is a constraint, the case of many embedded systems. As proved by the experimental results, in fact, test procedures turned out to be very compact and much more memory-efficient than conventional approaches relying on static sets of FPGA testing configurations to be stored in system memory.

Along the same line of research, the work in [16] revisited further concepts in the so-called application-dependent testing (ADT) for FPGA devices. The study presented in the paper identified a few limitations of state-of-the-art ADT approaches, which prevented a complete coverage for bridging faults and the practical applicability of the algorithms for test configuration generation. The work also introduced a set of new techniques that enabled us to overcome these limitations and effectively extend previous methodologies for ADT.

V. Security services

Recent advances in wireless technologies have enabled pervasive connectivity to Internet scale systems, including resource-constrained devices, such as mobile phones and tablets, a trend which has been referred to as ubiquitous computing over the past years. In particular, more and more security-critical applications are being deployed in such scenarios, making it crucial to provide security services to lightweight devices. Since security functions are typically based on computationally intensive cryptographic algorithms, deploying them is particularly challenging due to the limited computing power and other constraints typically affecting the above scenarios. Hence, in addition to hardware-related topics, the activity carried out by the author of this report also covered the implementation, deployment, and evaluation of security services in distributed environments. One of those services involved a particular security application, the so-called digital time stamping. The paper [17] describes the results of a research activity conducted cooperatively with an industrial party. The work involved a practical solution for the implementation of time stamping services and their exposition to the Internet for inter-enterprise integration. State-of-the-art time stamping algorithms and crucial issues related to their practical implementation were discussed. The focus was on integration problems which arise when a potentially large community of enterprises –relying on a handful of heterogeneous technologies– is willing to access remote third-party time stamping services. We proposed a practical architecture providing time stamping services, both in terms of *relative* temporal authentication using a linear linking scheme and *absolute* temporal authentication, based on publishing mechanisms and a trusted time source. The architecture was implemented using Web Services technologies. An integration experiment was conducted to evaluate the effectiveness of the proposed solution.

A similar application was addressed by [18] involving the design and implementation of an architecture for the provision of digital time stamping to mobile devices with limited resources. The architecture was described with respect to a case-study system and experimental results were also discussed.

A group of works addressed the interplay between security services and hardware acceleration. In particular, [19] discussed such challenges with respect to two key security services, Public-Key certification and digital time stamping, delivered to mobile devices. The work presented a multi-tier architecture combining a hardware-accelerated back-end and a Web Services based web tier for achieving interoperability while boosting performance. Further extending this research line, the work in [20] presented another solution combining hardware acceleration with a Web Services tier. The paper described the organization of the architecture, provided a detailed description of individual components, and presented the results of a thorough experimental campaign.

On the other hand, the activity described in [21] focused on a specific component, i.e. an FPGA-based key-store for improving the dependability of security services. A key-store is a facility for storing sensitive information, most typically the keys of a cryptographic application which provides a security service. In the paper, we presented a hardware implemented key-store, allowing secure storage and high performance retrieval of RSA keys. Since RSA is the most widely adopted standard for cryptographic keys, the proposed key-store can be effectively used to improve the dependability of a wide class of security services. The device was implemented on top of a Commercial Off The Shelf (COTS) Celoxica RC1000 board mounting a Xilinx Virtex-E 2000 FPGA part. We described the architecture of the hardware device, illustrated the organization of the associated device driver, and evaluated the security and performance gain achieved by integrating our device in real-world applications.

Since understanding the impact of the platform architecture is a key issue for deploying efficient security-enabled

applications on mobile devices, the work in [22] provided an experimental study of the influence that specific characteristics of mobile device platforms have on the final performance of security applications. The focus was on performance and resource utilization, which are key aspects when one develops applications on mobile devices. The case study was again a Web Services based solution for delivering public-key infrastructure services to mobile devices. Experiments were conducted on three different mobile terminals, spanning a large range of characteristics representative of resource-constrained devices. The results showed that: i) performance figures are not uniform in spite of similar underlying hardware characteristics, and ii) security and performance are often conflicting requirements.

As a further development of the activity, fostered by a collaboration of the author with a company providing monitoring and security services, the work in [23] presented an approach to the parsing of heterogeneous data streams, addressing scenarios where enterprise business processes are geographically distributed and involve entities in loosely coupled interactions. While cooperating, these entities generate transactional data streams, such as sequences of stock-market buy/sell orders, credit-card purchase records, Web server log entries, and electronic fund transfer orders. Such streams are often a collection of events stored and processed locally, and hence they typically have ad-hoc, heterogeneous formats. On the other hand, elements in such data streams usually share a common semantics and indeed they can be profitably mined in order to obtain combined global events. The above cited work, hence, introduced a solution relying on the definition of format-dependent grammars and automatic generation of ad-hoc parsers. The stream-dependent parsers can be obtained dynamically in a totally automatic way, provided that the appropriate grammar, written in a common format, is fed into the system. The work also presented a fully working implementation, that was successfully integrated into a telecommunication environment for real-time processing of billing information flows.

VI. Conclusions

This paper reported on the main research results achieved by the author, including activities carried out in the context of funded Research Projects, until year 2012. The most significant findings involved cryptographic hardware, the acceleration of cryptanalytical algorithms, new methodologies for design automation and testing addressing field programmable gate arrays, as well as the provisioning of security services in distributed environments. The report presented an exhaustive description of all the scientific works derived from the above activities, indicating the essential insights behind each of them and the main results collected from the experimental evaluation.

Acknowledgments

This report describes activities that have been partially funded by PRIN2005 project *COMMUTA: Hardware/software mutant components for distributed, dynamically reconfigurable systems*, and POR Campania FESR 2007-2013 project *Progetto Metadistretto del Settore ICT - Sistema di comunicazione per l'integrazione delle informazioni nella distribuzione commerciale e nei punti vendita*.

Following is a complete list of publications derived from the research activities described in this report.

References

[1] A. Cilardo, A. Mazzeo, L. Romano, and G. Saggese, "Carry-save Montgomery modular exponentiation on reconfigurable hardware," in *Proceedings - Design, Automation and Test in Europe Conference and Exhibition*, 2004, pp. 206–211.

[2] A. Cilardo, A. Mazzeo, L. Romano, and G. Saggese, "Architecture and FPGA implementation of a digit-serial RSA processor," in *New Algorithms, Architectures and Applications for Reconfigurable Computing*. Springer US, 2005, pp. 209–218.

[3] A. Cilardo, A. Mazzeo, L. Romano, and G. Saggese, "Exploring the design-space for FPGA-based implementation of RSA," *Microprocessors and Microsystems*, vol. 28, no. 4, pp. 183–191, 2004.

[4] A. Cilardo, L. Coppolino, N. Mazzocca, and L. Romano, "Elliptic curve cryptography engineering," *Proceedings of the IEEE*, vol. 94, no. 2, pp. 395–405, 2006.

[5] A. Cilardo, A. Mazzeo, N. Mazzocca, and L. Romano, "A novel unified architecture for public-key cryptography," in *Proceedings -Design, Automation and Test in Europe, DATE '05*, vol. 2005, 2005, pp. 52–57.

[6] A. Cilardo and N. Mazzocca, "Time efficient dual-field unit for cryptography-related processing," *IFIP Advances in Information and Communication Technology*, vol. 313, pp. 191–210, 2010.

[7] A. Cilardo, "A new speculative addition architecture suitable for two's complement operations," in *Proceedings -Design, Automation and Test in Europe, DATE*, 2009, pp. 664–669.

[8] A. Cilardo, A. Mazzeo, and N. Mazzocca, "Representation of elements in F_{2^m} enabling unified field arithmetic for elliptic curve cryptography," *Electronics Letters*, vol. 41, no. 14, pp. 798–800, 2005.

[9] A. Cilardo, "Efficient bit-parallel $GF(2^m)$ multiplier for a large class of irreducible pentanomials," *IEEE Transactions on Computers*, vol. 58, no. 7, pp. 1001–1008, 2009.

[10] A. Cilardo, "Exploring the potential of threshold logic for cryptography-related operations," *IEEE Transactions on Computers*, vol. 60, no. 4, pp. 452–462, 2011.

[11] A. Cilardo, L. Esposito, A. Veniero, A. Mazzeo, V. Beltran, and E. Ayguad, "A cellBE-based HPC application for the analysis of vulnerabilities in cryptographic hash functions," in *Proceedings - 2010 12th IEEE International Conference on High Performance Computing and Communications, HPCC 2010*, 2010, pp. 450–457.

[12] A. Cilardo, "The potential of reconfigurable hardware for HPC cryptanalysis of SHA-1," in *Proceedings -Design, Automation and Test in Europe, DATE*, 2011, pp. 998–1003.

[13] A. Cilardo, P. Durante, C. Lofiego, and A. Mazzeo, "Early prediction of hardware complexity in HLL-to-HDL translation," in *Proceedings - 2010 International Conference on Field Programmable Logic and Applications, FPL 2010*, 2010, pp. 483–488.

[14] A. Benso, A. Cilardo, N. Mazzocca, L. Miclea, P. Prinetto, and E. Szilrd, "Reconfigurable systems self-healing using mobile hardware agents," in *Proceedings - International Test Conference*, vol. 2005, 2005, pp. 468–476.

[15] A. Cilardo, N. Mazzocca, and L. Coppolino, "Virtual scan chains far online testing of FPGA-based embedded systems," in *Proceedings - 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, DSD 2008*, vol. 2008-January, 2008, pp. 360–366.

[16] A. Cilardo, C. Lofiego, A. Mazzeo, and N. Mazzocca, "Revisiting application-dependent test for FPGA devices," in *Proceedings - 16th IEEE European Test Symposium, ETS 2011*, 2011, p. 213.

[17] A. Cilardo, A. Mazzeo, L. Romano, G. Saggese, and G. Cattaneo, "Using Web Services technology for inter-enterprise integration of digital time stamping," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2889, pp. 960–974, 2003.

[18] A. Cilardo, D. Cotroneo, C. Di Flora, A. Mazzeo, L. Romano, and S. Russo, "Design and implementation of a high performance architecture for providing digital time stamping services to mobile devices," *Computer Systems Science and Engineering*, vol. 22, no. 3, pp. 103–112, 2007.

[19] A. Cilardo, L. Coppolino, A. Mazzeo, and L. Romano, "High-performance and interoperable security services for mobile environments," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3726 LNCS, pp. 1064–1069, 2005.

[20] A. Cilardo, L. Coppolino, A. Mazzeo, and L. Romano, "Combining programmable hardware and Web Services technologies for delivering high-performance and interoperable security," in *Proceedings - 15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing, PDP 2007*, 2007, pp. 381–386.

[21] A. Cilardo, A. Mazzeo, L. Romano, and G. Saggese, "An FPGA-based key-store for improving the dependability of security services," in *Proceedings - International Workshop on Object-Oriented Real-Time Dependable Systems, WORDS*, 2005, pp. 389–396.

[22] A. Cilardo, L. Coppolino, A. Mazzeo, and L. Romano, "Performance evaluation of security services: An experimental approach," in *Proceedings - 15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing, PDP 2007*, 2007, pp. 387–394.

[23] F. Campanile, A. Cilardo, L. Coppolino, and L. Romano, "Adaptable parsing of real-time data streams," in *Proceedings - 15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing, PDP 2007*, 2007, pp. 412–418.