IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 30, 2018, accepted February 20, 2019, date of publication March 14, 2019, date of current version April 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2904909

SDR Implementation of a D2D Security Cryptographic Mechanism

TITUS N. BALAN, (Member, IEEE), ALEXANDRA C. BALAN, (Member, IEEE), AND FLORIN D. SANDU, (Member, IEEE)

Department of Electronics and Computers, Transilvania University of Braşov, 500024 Braşov, Romania

Corresponding author: Titus N. Balan (titus.balan@unitbv.ro)

ABSTRACT Device-to-device (D2D) communication represents a promising technique to enable devices to communicate directly without the interaction of access points or base stations. The *ad hoc* and proximity nature of this communication introduce some very important security vulnerabilities. Key management, access control, privacy, secure routing, and transmission need dedicated signaling procedures and optimized implementation mechanisms that are appropriate for the mobile, low-energy, and low-processing power environment. This paper proposes a security mechanism for D2D communication involving the use of physically unclonable functions (PUF) for unique key generation, elliptic-curve cryptography (ECC) and Diffie–Hellman key exchange (DHKE) for key management, and Salsa20/20 as stream cyphering encryption method, suitable for confidentiality of the wireless transmissions. All these methods are implemented and tested on a software defined radio (SDR) communication platform consisting of a Zync-based system-on-chip (SoC), complemented by radio frequency (RF) daughter boards from analog devices–an integration using hardware and software co-design.

INDEX TERMS Device to device, Diffie-Hellman key exchange, encryption, elliptic-curve cryptography, key management, physically unclonable function, proximity services (ProSe), security, software defined radio (SDR), stream cyphering, Salsa20/20.

I. INTRODUCTION

D2D enables devices to communicate directly with each other without the involvement of fixed networking infrastructures such as Access Points or Base Stations. Though D2D can be applied using different technologies like Bluetooth, WiFi-Direct and Near Field Communications (NFC), an important standardization effort is represented by the 3GPP Proximity Services (ProSe) function.

D2D brings many advantages (shorter latency, decreased network traffic, power saving and a fallback system in the case of network failure), but it is generally acknowledged that the security of devices and data is a key factor for the success of Device to Device communication technology. No matter the technology used for D2D communications, the security problematic areas that must be considered are the same, including confidentiality, integrity, authentication, privacy, availability and dependability, as well as non-repudiation.

More security schemes were proposed for the D2D security mechanism, classified in the surveys of Wang and Zheng [1] or Hamoud *et al.* [2] with details on different methods for key management, authentication and access control. More than 40 research papers are mentioned, in these surveys, detailing security related work for D2D communication at different OSI layers. None of these solutions take into consideration the use of physical unclonable functions (PUF) for the unique identification of a device (integrated circuit) or as a method for generating private keys. Our implementation takes benefits of this PUF ''circuit fingerprinting'' methodology that is extremely suitable for mobile devices by implementing the mechanism on Static Random-Access Memory (SRAM)-based PUFs, that is achieving widespread adoption in commercial products, as some prototypes are already implemented by major handheld vendors like Intel, Samsung, UMC, Cypress, TSMC, IBM, Renesas. For instance, Samsung has released the ''Exynos I'' [3] dedicated Internet of Things (IoT) chip series with on-chip 'Security Sub-Systems' based on PUF, providing much higher levels of security compared to the conventional one-time-password based solutions [4].

The survey [2] also mentions whether the proposed D2D security mechanism are implemented or simulated and we

The associate editor coordinating the review of this manuscript and approving it for publication was Adrian Kliks.

have observed that many of the proposed solutions are just conceptual. Our algorithm proposal is also implemented in a prototype that will be detailed throughout the paper. There are more security issues needed to be taken into consideration for a D2D security mechanism, like confidentiality, integrity and authentication.

Our proposed security scheme offers solutions for the following problems:

- key management using RO-PUF for the unique secret key generation, Elliptic Curve Cryptography for generating the public key corresponding to the secret key, Diffie Hellmann for generating the shared secret key and key exchange;
- data encryption using stream cyphering method based on Salsa20-20 algorithm, suitable for real-time communications encryption;

All these implementations are performed on a Software Defined Radio (SDR) [5] communication platform consisting of a Zync based System on Chip (SoC) on a Digilent "ZedBoard" [6] complemented by radio frequency (RF) daughter-boards from Analog Devices (FMCOMMS3 and FMCOMMS4) [7], an integration solution using hardware and software co-design. The SDR platform represents an excellent prototyping system, allowing the possibility to implement in hardware the complex security algorithms. It offers the flexibility to apply the designed modules (Intellectual Property - IP cores) in different radio communication solutions, from WiFi to LTE - either a custom approach or one that includes industry proven sub-systems based on MATLAB, Xilinx Vivado or GNU-Radio.

The paper is organized as follows: The first section describes the D2D security issues and the mechanisms that should be implemented relative to the 3GPP description of the ProSe function. We will mention the solutions and motivation for the selection of the algorithms to be part of our prototype, compared to other related work implementations. The 2nd section details the proposed security algorithms and their FPGA hardware implementation using Xilinx Vivado and the own written IP cores. The 3rd section is detailing our demonstrator setup for the security scheme using the implemented elements and the integration between the cryptographic and communication modules, respectively the ARM and FPGA co-existence. Next, results of the implemented algorithms are presented, as well as the possibility to use the presented security methodology in order to enhance the existing key exchange mechanism of the ProSe 3GPP standard. The last paragraph presents our conclusions.

II. SECURITY CONSIDERATIONS FOR D2D COMMUNICATIONS AND RELATED WORK

The D2D communication can occur either on operator's licensed spectrum (underlying 3GPP LTE-A networks) or in the unlicensed spectrum (Bluetooth, WiFi-Direct).

While the D2D communication is mostly using the Industrial, Scientific and Medical (ISM) spectra and works in a pure autonomous way, the 3GPP specifications ProSe (Proximity Services) have some well-established signaling procedures, as well as an own key management method [8]. First introduced in Release 12 of the 3GPP specifications, the LTE-A ProSe relies on multiple enhancements to existing LTE standards including new functional elements and a "sidelink" air interface for direct connectivity between devices. There are three main scenarios for D2D communications (Fig.1):

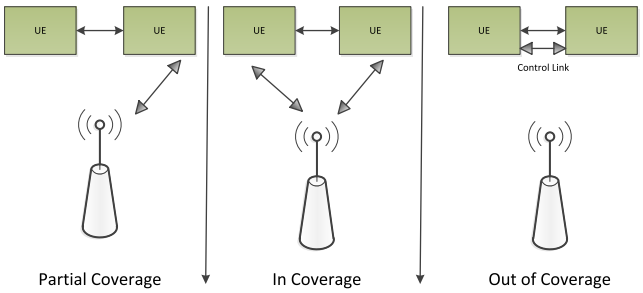


FIGURE 1. D2D communication scenarios.

- the In-Coverage scenario in which the devices are in the coverage area of a base station (BS) / access point (AP), traffic offloading being the most common use-case
- the Relay Coverage in which part of the devices are out of base station coverage (Partial Coverage), but they can communicate through relaying their communication data via other covered devices
- the Out-of-Coverage case when the network coverage is absent. A typical use-case is the Public Safety communication. Devices can autonomously set up connections and start D2D communications with each other in their proximity, without the assistance of any operators.

Each of the scenarios above have their own security schemes, some of them better established via the 3GPP ProSe standardization. One of the most challenging D2D scenario is the out of coverage scenario, when the communication is done autonomously - in this situation the ProSe scenario is similar to the unlicensed ISM case, and the same security methods can be applied.

Most of the existing literature refers to this 3rd scenario, detailing methods for key management, authentication, access control for spontaneous and non-assisted D2D communication.

Several D2D key management solutions are detailed in other research papers, taking benefit of Diffie Hellman Key Exchange (DHKE): Shen *et al.* [9] detail a method for establishing a shared secret key between two D2D devices based on DHKE, Zhang *et al.* [10] present a method to realize a session key agreement between two D2D devices under the control of BS, while Ekberg *et al.* [11] detail the usage of DHKE for setting a security association, followed by mutual authentication via home core network certificates. We have also used the Diffie-Hellman algorithm, but we have complemented it with Elliptic Curve Cryptography (ECC) for

generating the public key corresponding to the secret key. ECC is a public key encryption technique based on elliptic curves theory that can be used to create faster, smaller, and more efficient cryptographic keys than other first-generation encryption public key algorithms such as RSA and Diffie-Hellman, so it is recommended for mobile and wireless devices [12]. Another method proposed by related research papers is the key management based on Attribute-Based Encryption (ABE), a type of public-key encryption in which the secret key of a user and the cipher-text are dependent upon attributes [13].

For authentication purpose, several methods were detailed in research papers dedicated to D2D implementations: Hash Message Authentication Code - HMAC [10], sharing pin authentication [13] or certificate authentication [11], [14].

We have observed that there are no D2D security schemes taking benefit of the device uniqueness model based on physical unclonable functions (PUF). There are several methods for authentication or key management using the PUF [15], but not applied on D2D key management schemes. Considering that the chip manufacturers - including mobile devices vendors - started including PUF security in their chips, we hereby propose the usage of PUFs in D2D security schemes.

When it comes to data encryption for real data streaming, it is known that stream cyphering is more suitable compared to block cyphering, as it was also used in GSM networks (GEA methods based on the KASUMI algorithm), many of the available solutions being based on block Cyphering. The 3GPP ProSe function uses the EEA (extended Euclidean algorithm) cryptography - while the version EEA-2 uses the 128-bit AES (Advanced Encryption Standard), more recently, EEA-3 has become available, based on the ZUC [16] stream cipher [17].

We are proposing the usage of Salsa20/20 in our implementation, a stream cyphering method very efficient and considered secure, as it was not yet compromised so far.

The surveys in [1], [2], and [18] describe - besides some other work related to D2D security - some types of attacks that can be performed on D2D communication. A single compromised node can be turned into a malicious one that brings down a complete system or can cause disasters.

III. PROPOSED D2D KEY MANAGEMENT AND ENCRYPTION MECHANISM

The algorithms used for D2D security are presented in Figure 2. RO PUF (Ring Oscillator Physical Unclonable Functions) implemented on Zynq are used to generate a secret key for one device involved in D2D communications. Each device involved in the communication gets a secret key generated with RO PUF circuits. Elliptic Curve Cryptography operations are used for the generation of a public key corresponding to the secret key generated with RO PUF. The next step is to generate a shared secret key for each device using: i) the secret key generated with RO PUF; ii) the ECC cryptographic operations and iii) the public key of other device involved in the D2D communication.

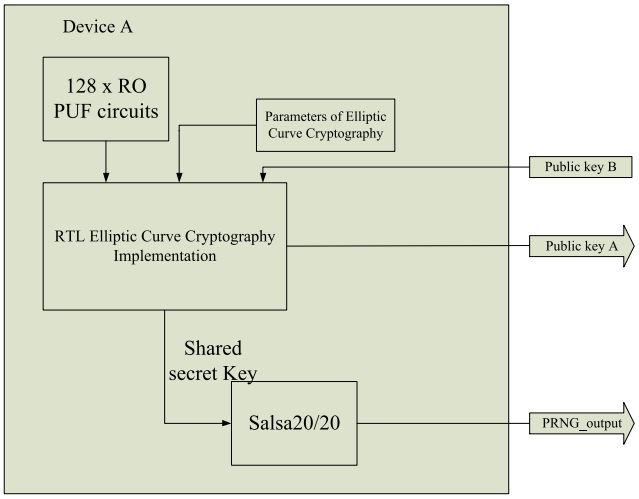


FIGURE 2. Security mechanism implemented for devices involved in D2D communication.

It has been used the Diffie-Hellman anonymous key agreement protocol that allows two parties, each having an elliptic curve public-private key pair, to establish a shared secret key over an insecure channel. The shared secret key is used as a seed into Salsa20/20 pseudo-random generator in order to have a stream of secret keys to be used in a symmetric encryption.

The encryption scheme for the D2D communication between Device A and Device B has the following phases:

i) Generate a secret key with RO PUF circuits for each device. Due to the process variations, each secret key will be unique for each device;
ii) Generate the public key for each device using the corresponding RO PUF secret key and ECC operations;
iii) Switch the public keys between device A and Device B (Device A sends his Public Key A and receives the Public Key B);
iv) Compute the shared secret key for Device A and Device B (in case of Device A there are used: the corresponding RO PUF secret key, the Public key from Device B and the ECC operations);
v) Use the shared secret key as a seed for Salsa20/20 algorithm and generate pseudo random keys;
vi) Use the XOR operation between the Salsa20/20 PRNG (pseudorandom number generator) output and the plain text message in order to encrypt the information

IV. IMPLEMENTATION

A. RO PUF CIRCUITS IMPLEMENTED ON ZYNQ

Inspired by biometrics, PUFs provide a unique way to identify integrated circuits. Comparable in a simplistic way with a "unique fingerprint" of an IC - that differentiates one IC from another (though apparently identical) - PUFs exploit the inherent variability in IC manufacturing to implement challenge-response functions whose outputs depend on the inputs and on the physical micro-structure of the devices.

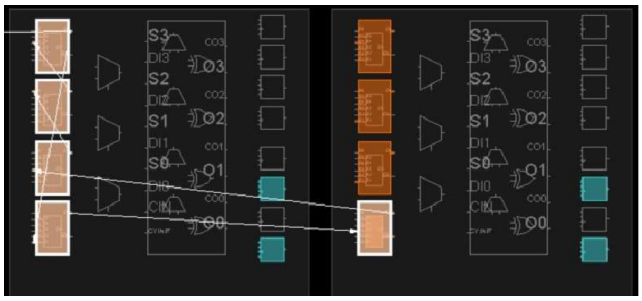


FIGURE 3. Five inverters connected in a loop.

The secret key is generated using 128 RO PUF circuits which exploits IC manufacturing process natural variations in attributes of the transistors (length, width, oxide thickness).

One RO PUF is composed of 5 inverter gates connected in a loop as may be seen in Figure 3.

The inverters are manually placed and routed on the hardware resources using constraints like the ones described in Figure 4.

```
set_property BEL A6LUT [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst/nand2/a2_INST_0}]
set_property LOC SLICE_X88Y58 [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst/nand2/a2_INST_0}]
set_property BEL A6LUT [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst0/nand2/a2_INST_0}]
set_property LOC SLICE_X88Y59 [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst0/nand2/a2_INST_0}]
set_property BEL A6LUT [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst1/nand2/a2_INST_0}]
set_property LOC SLICE_X88Y60 [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst1/nand2/a2_INST_0}]
set_property BEL A6LUT [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst10/nand2/a2_INST_0}]
set_property LOC SLICE_X88Y61 [get_cells {ring_osc_gen[58].ring_osc_inst/ro_puf_inst10/nand2/a2_INST_0}]
```

FIGURE 4. RO PUF constraints for manually place and route.

Those inverters connected in a loop generate a periodical signal. Due to process variations the frequencies of two generated signals are slightly different. The periodical signal is used as a clock signal for a 13-bit width counter. Considering two counters each with a clock signal generated by 5 inverters connected in a loop, one of the counters will reach first the maximum value - due to the above-mentioned process variations that occur during the manufacturing process. Using a comparator for the output of the two counters, the comparator output will be unpredictable (0 or 1), representing the RO PUF response, 1 bit from the total 128 bits used as a secret key.

One RO PUF instance used in our implementation is composed of the digital circuits mentioned in Figure 5.

At first, 64 bits are generated using clock signals produced by the ro_puf_inst and ro_puf_inst1 and then the clock signals are switched with the ones generated with ro_puf_inst0 and ro_puf_inst10. In this manner, the number of 13-bit width binary counters is decreased twice - 64 instead of 128. Figure 5 shows an instance of RO PUF which generates the 50 bit and 100 bit from 128 bits. There are a total of 64 instances of RO PUFs as the one presented in Figure 5.

The statistical analysis of RO PUF circuits regarding their unicity and reproducibility is beyond the scope of this paper and has been extensively analyzed in [19]–[21].

Because of the FPGA routing complexity and limitations, few existing PUF circuits can be implemented on FPGA. After many attempts of PUF implementations we concluded

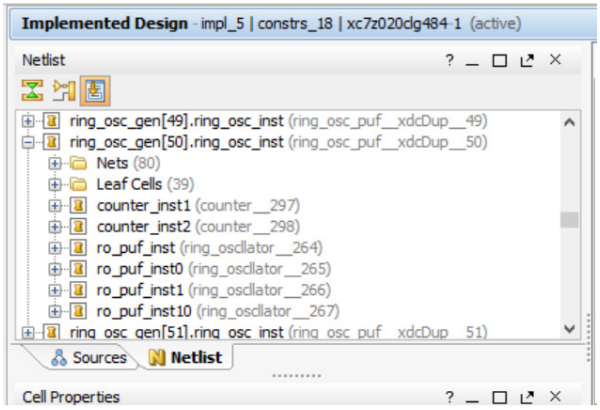


FIGURE 5. RO PUF instance no. 50 – hardware resources.

that the well-known ring oscillator PUF and the Latch based PUF are appropriate for an FPGA application. However, in case of an ASIC integrated circuit, the SRAM PUF is a suitable choice.

B. ELLIPTIC CURVE CRYPTOGRAPHY

Symmetric key cryptosystems use the same key for encryption and decryption. Having the disadvantage of needing a secret key known by all the participants (a secret shared key) they have, nevertheless, the advantage of a reduced computing time [22]. Therefore, we will use symmetric key cryptosystems for encryption/decryption and we will generate a secret shared key.

Compared to other encryption technologies, ECC is helpful for use in low-memory and low-computing environments such as mobile devices and wireless devices. For example, a 160-bit ECC encryption key provides the same security as a 1024-bit RSA (Rivest-Shamir-Adleman) encryption key and can be up to 15 times faster, depending on the platform on which it is implemented [23], [24].

In this section there are illustrated details and results of the Elliptic Curve Cryptography implemented in hardware using HDL (Hardware Description Languages). For fast and accurate arithmetic in hardware implementations, elliptic curves over binary field $\boldsymbol{F}_{2^m}$ (where m is a positive integer) are used. An elliptic curve E over the finite field $\boldsymbol{F}_{2^m}$ is given through the following equation.

$$y^2 + xy = x^3 + ax^2 + b, \quad \text{where } x, y, a, b \in F_{2^m}$$

The points on E are denoted as:

$$E(F_{2^m}) = \left\{(x, y): x, y \in F_{2^m} \text{ satisfy } y^2 + xy == x^3 + ax^2 + b\right\}$$

The public key of each entity is computed as elliptic curve scalar multiplication. We will produce a "trapdoor function" where the special information or "trapdoor" is the ID value generated using PUF circuits. In order to implement the elliptic curve scalar multiplication, the following information is required [25]:

TABLE 1. Elliptic curve cryptography parameters.

Finite field F_{2^m}	Values
m	163
Irreducible polynomial	$p(t) = t^{163} + t^7 + t^6 + t^3 + 1$
Elliptic curve - polynomial basis	Values
a	1
G_x	2 fe13c053 7bbc11ac aa07d793 de4e6d5e 5c94eee8
G_y	2 89070fb0 5d38ff58 321f2e80 0536d538 ccdaa3d9

Given a curve E defined along an equation in a finite field (such as: $y^2 = x^3 + ax + b$), point multiplication is defined as the repeated addition of a point along that curve.

Let $nP = P + P + \ldots P$ for a scalar (integer) n and a point $P = (x,y)$ that lies on the curve E. The security of modern ECC depends on the intractability of determining n from $Q = nP$ given the known values of Q and P. This is known as the elliptic curve discrete logarithm problem.

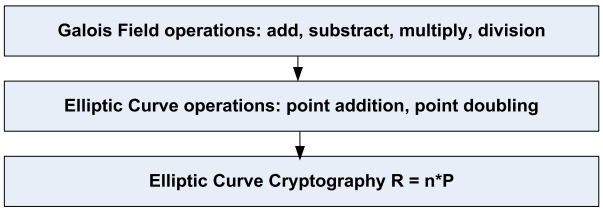


FIGURE 6. Elliptic curve cryptography - layers.

We decomposed the problem in three layers as it can be seen in Figure 6.

The first layer implements the operation over the Galois Field (GF): addition, subtraction, multiplication and multiplicative inverse. The field $GF\ (2^n)$ is defined by a set of 2^n unique elements that is closed under both addition and multiplication, in which every non-zero element has a multiplicative inverse and every element has an additive inverse. As with any field, addition and multiplication are associative, distributive and commutative. The field $GF\ (2^n)$ is defined over an irreducible polynomial of degree n with coefficients in $GF\ (2^n)$. The primitive polynomial has a root α, named primitive root where $\alpha^{2^n-1} = 1$ and α^i, where i $< 2^n - 1$ generates a different element from $GF\ (2^n)$. The Galois field $GF\ (2^n)$ may be represented by the set of all polynomials of degree at most $n - 1$, with binary coefficients, as can be seen in Table 2.

In our implementation, the Galois Field elements are considered in binary vector representation. The addition and subtraction operations in hardware over F_{2^m} are simple bitwise XOR operations. Multiplication in a finite field is multiplication modulo an irreducible reducing polynomial used to define the finite field.

TABLE 2. Galois elements – different representations.

Elements	Polynomials	Binary vectors
α^{127}	$\alpha^6 + \alpha^5 + \alpha^2 + \alpha$	01100110
α^{128}	$\alpha^7 + \alpha^6 + \alpha^3 + \alpha^2$	11001100
α^{131}	$\alpha^5 + \alpha^3 + \alpha^2 + 1$	00101110

TABLE 3. Galois field operations.

Operand 1	164'h2fe13c0537bbc11acaa07d793de4e 6d5e5c94eee8
Operand 2	164'h0000000004587684f96c8dd6558 e69339a1efed9
Addition	164'h2fe13c0537fe46728536b5a458bc0 046dc68a1031
Multiplication	164'h5c57994079089d855a62060851eb 4d973daa131a3
Division/Multi p licative	164'h63f514f39f4587684f96c8dd6558e 69339a1efed9

We implement the multiplication as a multiplication followed by division using the reducing polynomials as the divisor.

First, we consider a general algorithm for division of two binary polynomials. In the second attempt for multiplication we consider a multiplication followed by a particular division using a fix (known) value of the reducing polynomial as the divisor. For the multiplicative inverse we implement the extended Euclidian algorithm (EEA) which is based on polynomial division and multiplication over Galois Field. In order to optimize the extended Euclidian algorithm in terms of area we optimized the polynomial division algorithm. In our implementation we considered the representation of the Galois Field as binary vectors. Table 3 shows an example of operations in Galois Fields over $F_{2^{163}}$ with the generator polynomial

$$p\,(t) = t^{163} + t^7 + t^6 + t^3 + 1$$

The second layer contains the operations of elliptic curve cryptography: point addition and point doubling, which are based on the operations from the first layer. The equations for point addition and point multiplication are given below:

$R = P + Q$	$R = 2P$
$x_R = \lambda^2 + \lambda + x_p + x_q + a$	$x_R = \lambda^2 + \lambda + a$
$y_R = \lambda\left(x_p + x_R\right) + x_R + y_p$	$y_R = x_P^2 + \lambda x_R + x_R$
$\lambda = \dfrac{y_Q + y_P}{x_Q + x_P}$	$\lambda = x_P + \dfrac{y_P}{x_P}$

TABLE 4. Point addition and point doubling.

P	P_X=164'h2fe13c0537bbc11acaa07d793de4e6d5e5c94eee8 P_y=164'h63f514f39f4587684f96c8dd6558e69339a1efed9
Q	Q_X=164'h6c6827cbef986f34a18e3b880fb362721468cc0fd Q_y=164'h6c3e54aa8914d1e70e28520b5601a488efd44d65e
Point Addition $R=P+Q$	R_x=164'h1e81dc792e6eeb0296a809110afd67f9a473c0916 R_y=164'h564c446ed70fd5de82ca065dd0890a7072805cf68
Point Doubling $R=2P$	R_x=164'h6c6827cbef986f34a18e3b880fb362721468cc0fd R_y=164'h6c3e54aa8914d1e70e28520b5601a488efd44d65e

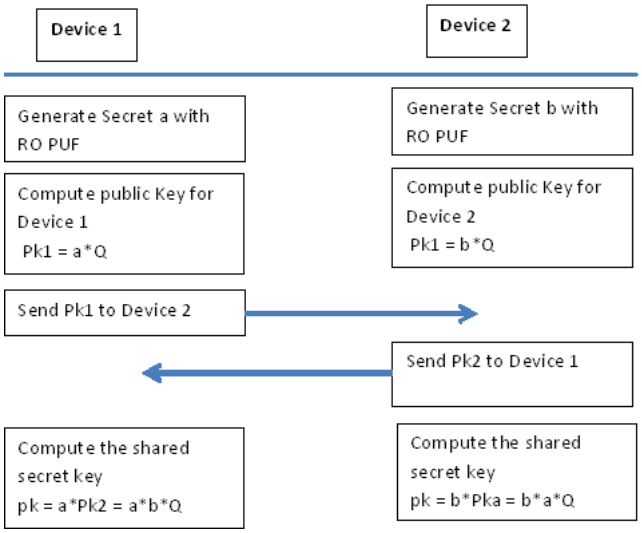


FIGURE 7. Diffie Hellman key agreement protocol based on elliptic curve cryptography.

where $\lambda, x_p, x_q, y_p, y_q, x_R, y_R$ are elements of finite field $F_{2^{163}}$. In order to obtain an optimal implementation, in terms of area and speed, for these operations we must have an optimal implementation of operations in finite fields $F_{2^{163}}$.

C. DIFFIE HELLMAN KEY EXCHANGE

Elliptic curve Diffie-Hellman is an anonymous key agreement protocol that allows two parties, each having an elliptic curve public-private key pair, to establish a shared "secret" over an insecure channel [26]. A schematic description of the Diffie Hellman key agreement protocol is presented in Figure 7.

The Diffie Hellman key agreement protocol has the following steps: i) Each device generates his own secret based on RO PUF circuit responses; ii) Using the generated secret key and the elliptic curve cryptography described in Section A the public cryptographic key is produced; iii) the public keys are exchanged between devices; iv) the shared secret key is computed using: public key, secret key generated with RO PUF and ECC operations.

FIGURE 8. Experimental setup – Back-to-back "Zedboards" with FMCOMMSx AD daughterboards.

V. EXPERIMENTAL SETUP

As experimental setup we have used the SDR prototyping environment offered by the Digilent "Zedboard", combining the Xilinx Zynq®-7000 All-Programmable SoC (ARM® dual-core Cortex™-A9 + 28 nm programmable logic) with the Analog Devices AD-FMCOMMS3-EBZ FMC (FPGA Mezzanine Card) module featuring the AD9361 integrated RF Agile Transceiver.

The Zynq-7000 SoC (System on Chip) offers the possibility to combine the software programmability of an ARM-based processor with the hardware reconfigurability of a FPGA, enabling hardware acceleration while integrating CPU, DSP and mixed signal functionality on a single device. The features listed above make the Zynq7000 a good platform for Software Defined Radio (SDR) implementations of a broad range of transceiver applications for wireless communications. We recommend it as very suitable for experimenting Device-to-Device communication and integration with complex security implemented functions. For our demonstrator we have used a back-to-back configuration (Figure 8) with direct coupled radio interfaces, without any limitations and interferences with the public spectrum.

A description of the environment setup on Zedboard and the usage of the ARM core using a Linux operating system was presented by Dustinta and Stanciu [27].

In order to set-up the SDR work environment we have used the Linux ARM co-processor and the SoC implementation using Xilinx Vivado software [28] at FPGA level,

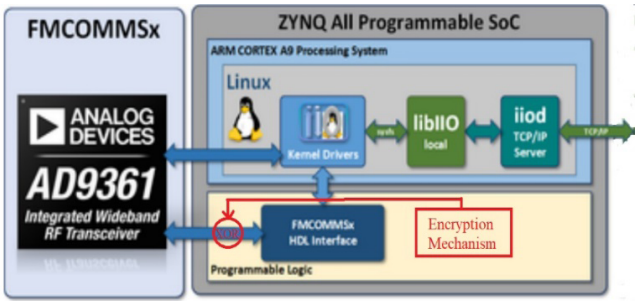


FIGURE 9. Encryption mechanism alongside the Zynq programmable SoC.

connected with the Analog Devices AD-FMCOMMS3-EBZ board as radio interface. The communication sub-system was implemented using the AD IP Core for wireless communication.

Development for combining the communication modules with our implemented security modules can be performed in different ways:

- Software implementation - running on the ARM core Linux implementation provided by Analog Devices - that also instantiates the communication with the AD Communication IP Core (see Figure 9). On top of this operating system, other communication software packages can be used (for example those from the open source GNU Radio).
- Software and hardware co-design using MATLAB - particularly the MATLAB Communication module and the belonging LTE Advanced D2D communication modules [29]
- Custom using the AD IP Core via Xilinx Vivado and running C code on top of this IP Core

When it comes to implementation of the complex cryptographic modules, because of the increased processing power needed, these will run in the FPGA part, as we have also implemented them on the Zync platform. We have implemented different IP cores with the cryptographic functions mentioned above – as depicted in Figure 9. The encryption mechanism is implemented in hardware alongside the AD communication IP core FMCOMMSx. Due to the fact that we have chosen symmetric encryption based on XOR operation, the performance of the transmission is not affected. Instead of sending plain-text data from the FMCOMMs HDL interface, an XOR between data and a pseudo random key resulting in encrypted data is transmitted.

VI. IMPLEMENTATION RESULTS

A. RO PUF CIRCUITS

The 128 bits secret key generation with RO PUF circuits is presented in Figure 10. Results are visualized with the ChipScope Logic Analyzer provided by Xilinx Vivado. Firstly, there are generated 64 bits representing the LBSs of the secret key and the other half of it by changing the ring oscillators composed of 5 inverters connected in a loop.

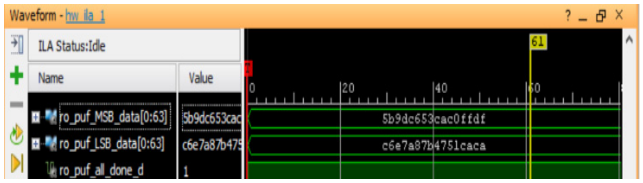


FIGURE 10. RO PUF circuits results implemented on Zynq.

TABLE 5. Hardware resource usage for ECC over $F_{2^{163}}$.

	Number of 4 Input LUTs	Number of occupied Slices	Number of Slice Flip Flops	Frequency
Elliptic curve cryptography	26241	10536	19797	114Mhz

TABLE 6. ECC – number of clock cycles.

n	164'h1111..11
$R = nP$	R_x=164'h573a9d414527f7b1b77fe988a26f64c494aa46ba8 R_y=164'h08de4ba2bbfe587a51b730eb3ad49ec5bc2875920
No. of clock cycles	12 834 130
Total time	$\frac{1}{freq} * No\ of\ clock\ cycles = \frac{1}{114 * 10^6} * 12\ 834\ 130$ $= 0{,}1002\ s = 100\ ms$

B. HARDWARE IMPLEMENTATION OF ECC

This section is summarizing the results of elliptic curve cryptography implemented in Verilog and synthesized, placed and routed with Xilinx Vivado.

Table 5 presents the implementation results of elliptic curve cryptography over $F_{2^{163}}$, in terms of hardware resource usage and frequency. Table 6 presents the results with the parameters given in Table 1.

C. EXPERIMENTAL SETUP – BACK-TO-BACK "ZEDBOARDS"

In Figure 8 it was presented the connection between two devices: i) Device A represented by a "Zedboard" development platform and FMCOMMS4 [30] and ii) Device B which is implemented on a Zedboard platform and FMCOMMS3 [31]. Device A is running a GNU application over a Linaro operating system that receives the encrypted data from Device B. Details regarding this implementation are presented in [27]. Device B is using the HDL reference design presented in [32], the version with ARM microprocessor which runs a C code in order to transmit encrypted data. The C code is available on [33] and the encryption of data was done with the above-detailed chosen algorithms.

D. PROPOSAL FOR 3GPP ProSe KEY MANAGEMENT FUNCTION ENHANCEMENT FOR AN "OUT-OF-COVERAGE SCENARIO"

As part of the key management signaling in LTE-A, security parameters are provided by a network node called the ProSe Key Management Function. This node may physically be part of the ProSe Function included in the LTE-A core network. Central to the security is the ProSe Group Key (PGK) parameter. It is used as a basis to derive input parameters for the security algorithm. Each PGK is provided with an expiry time. By providing the UE (User Equipment) with PGKs valid for different times, the UE may operate for a longer time without further parameter provisioning from the core network – like, for example in the out-of-coverage – always taking a PGK valid for the actual time [34]. However, this method of operation has some disadvantages mainly for devices that were out of coverage for a long time, so a possible solution would be for the device to generate locally some parameters using the PUF generated secret key.

VII. CONCLUSION

This paper proposed a D2D security mechanism for key management and data encryption, implemented and tested on two Digilent "Zedboard" FPGA based systems. The security mechanism is generic, can be applied to any type of communication (Wifi Direct, Bluetooth) but it can improve also the standardized 3GPP ProSe Key Management Function in the out-of-coverage scenario.

Although most of the papers approaching D2D cryptographical methods are focused on describing the algorithms and their advantages in a theoretical or simulated manner, we have implemented the proposed security algorithms in the Verilog HDL.

Important results of the research work include:

- A proposed security mechanism for D2D communication involving some novel solutions like PUF-based key generation, efficiency of ECC as public key generation and a stream cyphering encryption method using Salsa20/20, suitable for confidentiality of the wireless transmissions.
- Actual implementation of the algorithms, not only in software, but also hardware-accelerated on the Zync SoC platform with the Analog Devices RF daughter-boards.
- A method for implementing a prototyping environment of the D2D communication and security by usage of dedicated SDR platforms, with a HW-SW codesign that can be used for future research and development.

The implementation of the RO PUF circuit on the Zync SoC for unique secret key generation can prevent impersonation attacks or other unsecure methods for nonce or seed generation in cryptographic functions. We consider that dedicated circuits including PUFs are perfect identifiers for any hardware element and will become very popular, so the proposed methods are valuable for many future areas of applicability in embedded systems, from mobile devices to IoT and Vehicle-to-anything V2X communications.

The paper was focused on the security related implementation and the integration of the SDR configurations, but the methods we have accomplished for communications can be adapted and enhanced by deployment on different radio technologies (some of them facilitated, for instance, by predefined LTE-A MATLAB routines, integrated with Simulink and Analog Devices specific drivers).

Further research can extend the presented approach on the integration of signaling related to device discovery procedures or on the side-link channels and transmissions.

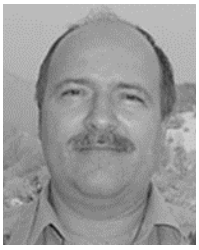
REFERENCES

[1] M. Wang and Y. Zheng, "A survey on security in D2D communications," *Mobile Netw. Appl.* vol. 22, no. 2, pp. 195–208, 2017.
[2] O. N. Hamoud, T. Kenaza, and Y. Challal, "Security in device-to-device communications: A survey," *IET Netw.*, vol. 7, no. 1, pp. 14–22, Jan. 2018.
[3] *Samsung Press*. Accessed: Nov. 2018. [Online]. Available: https://www.samsung.com/semiconductor/minisite/exynos/newsroom/pressrelease/samsung-begins-mass-production-of-its-first-iot-designated-exynos-i-t200/
[4] *The Reliability of SRAM PUF*. Accessed: Nov. 2018. [Online]. Available: https://www.intrinsic-id.com/wp-content/uploads/2017/08/White-Paper-The-reliability-of-SRAM-PUF.pdf
[5] F. K. Jondral, "Software-defined radio–basics and evolution to cognitive radio," *EURASIP J. Wireless Commun. Netw.*, vol. 3, pp. 275–283, Dec. 2005.
[6] *Digilent Zedboard Reference Manual*. Accessed: Nov. 2018. [Online]. Available: https:// reference.digilentinc.com/reference/programmable-logic/zedboard/reference-manual
[7] Analog Devices. *FMCOMMS Users Guide*. Accessed: Nov. 2018. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms3-ebz
[8] *Proximity-Based Services (ProSe)–Security Aspects*, document TS 33.303, 2017.
[9] W. Shen, W. Hong, X. Cao, B. Yin, D. M. Shila, and Y. Cheng, "Secure key establishment for device-to-device communications," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 336–340.
[10] A. Zhang, J. Chen, R. Q. Hu, and Y. Qian, "SeDS: Secure data sharing strategy for D2D communication in LTE-Advanced networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2659–2672, Apr. 2016.
[11] J.-E. Ekberg, M. A. Uusitalo, and Z. Li, "Device to device communication security," WO Patent 2 014 207 506 A1, Dec. 31, 2014.
[12] V. Ajithkumar and K. Satyanarayan, "A survey on security of mobile handheld devices through elliptic curve cryptography," *Accents Trans. Inf. Secur.*, vol. 2, no. 6, pp. 2455–7196, 2017. doi: 10.19101/TIS.2017.26001.
[13] H. Kwon, C. Hahn, D. Kim, K. Kang, and J. Hur, "Secure device-to-device authentication in mobile multi-hop networks," in *Proc. 9th Int. Conf. Wireless Algorithms, Syst., Appl.*, Jun. 2014, pp. 267–278.
[14] Y. Jung, E. Festijo, and M. Peradilla, "Joint operation of routing control and group key management for 5G ad hoc D2D networks," in *Proc. Int. Conf. Privacy Secur. Mobile Syst. (PRISMS)*, May 2014, pp. 1–8.
[15] U. Chatterjee *et al.*, "Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database," *IEEE Trans. Depend. Sec. Comput.*, to be published. doi: 10.1109/TDSC.2018.2832201.
[16] *Confidentiality and Integrity Algorithms EEA3 & EIA3; Document 1: EEA3 and EIA3 Specifications*, document 3GPP TS 35.221, Version 14, 2017.
[17] *System Architecture Evolution (SAE); Security Architecture*, document 3GPP 33.401, 2016.
[18] M. Haus, M. Waqas, A. Y. Ding, Y. Li, S. Tarkoma, and J. Ott, "Security and privacy in device-to-device (D2D) communication: A review," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1054–1079, 2nd Quart., 2017.
[19] A.-R. Sadeghi, D. Naccache, and P. Tuyls, *Towards Hardware-Intrinsic Security*. Berlin, Germany: Springer, 2010.
[20] A. Stanciu, M. N. Cirstea, and F. D. Moldoveanu, "Analysis and evaluation of PUF-based SoC designs for security applications," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5699–5708, Sep. 2016.

[21] Y. Bendavid, N. Bagheri, M. Safkhani, and S. Rostampour, "Iot device security: Challenging 'a lightweight RFID mutual authentication protocol based on physical unclonable function," *Sensors*, vol. 18, no. 12, p. 4444, 2018.
[22] M. Schramm, D. Reiner, and M. Heigl, "A vendor-neutral unified core for cryptographic operations in GF(p) and GF(2^m) based on montgomery arithmetic," *Secur. Commun. Netw.*, May 2018, Art. no. 4983404.
[23] R. Afreen and S. C. Mehrotra. (2011). "A review on elliptic curve cryptography for embedded systems." [Online]. Available: https://arxiv.org/abs/1107.3631
[24] Infosec Institute. *ECC: A Case for Mobile Encryption*. Accessed: Nov. 2018. [Online]. Available: https://resources.infosecinstitute.com/ecc-case-mobile-encryption/#gref
[25] Certicom Research. (2010). *SEC 2: Recommended Elliptic Curve Domain Parameters*. [Online]. Available: http://www.secg.org/SEC2-Ver-1.0.pdf
[26] A. Roy, "Brief comparison of RSA and Diffie–Hellman (public key) algorithm," *Accents Trans. Inf. Secur.*, vol. 1, no. 1, pp. 28–30, 2016.
[27] D. Duştinţă and A. Stanciu, "System on chip development platform for software defined radio," *Rev. Air Force Acad.*, vol. 1, pp. 65–70, May 2018.
[28] Xilinx Vivado Design Suite. *User Guide-Design Flows Overview, UG892 (v2018.2) UG892 (v2018.3)*. Accessed: Dec. 5, 2018. [Online]. Available: https://www.xilinx.com/support/documentation/sw_manuals/xilinx2018_3/ug892-vivado-design-flows-overview.pdf
[29] Mathworks Application Work. *Device-to-Device (D2D) Communication for Public Safety in LTE*. Accessed: Sep. 2018. [Online]. Available: https://uk.mathworks.com/campaigns/offers/device-to-device-communication-for-public-safety-in-lte.html
[30] *AD-FMCOMMS4 User Guide*. Accessed: Sep. 2018. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms4-ebz
[31] *AD-FMCOMMS3 User Guide*. Accessed: Sep. 2018. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms3-ebz
[32] *AD9361 HDL Reference Designs*. Accessed: Sep. 2018. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz/reference_hdl
[33] *AD9361 No-OS Software*. Accessed: Sep. 2018. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz/software/baremetal
[34] Rohde & Schwarz Application Note. *1MA264: Device to Device Communication in LTE*. Accessed: Sep. 2018. [Online]. Available: https://www.rohde-schwarz.com/applications/device-to-device-communication-in-lte-white-paper_230854-142855.html

TITUS N. BALAN was born in Braşov, Romania, in 1984. He received the Dipl. Eng. degree from the Electrical Engineering and Computer Science Faculty, Transilvania University of Braşov, Romania, in 2007, the M.Sc. degree in electronics engineering and telecommunications from the Transilvania University of Braşov, and the Ph.D. degree from the Electrical Engineering and Computer Science Faculty, Transilvania University of Braşov, in 2011. His Ph.D. dissertation was on mobility solution in heterogeneous networks. He was the Product Line Manager with Siemens Communications Media and Technology. He was involved in research and development activities for long-term evolution (LTE) mobile communication networks with the Dusseldorf Research and Development Center, Nokia Siemens Networks. Since 2013, he has been a Senior Lecturer with Transilvania University and also an In Charge of the Cybersecurity Master Study Program. He is currently a Senior Expert with the ATOS Big Data and Security Division, responsible for solution definition for research and development projects and presales in commercial projects in the area of communication solutions, security, big data, and mission critical communications. His research interests include telecommunications, computer networks, cybersecurity, the Internet of Things (IoT), and cloud and virtualization. He is a member of the European Union IRACON Cost Action CA15104.

ALEXANDRA C. BALAN was born in Braşov, Romania, in 1988. She received the Dipl. Eng. degree in computers and information technology and the M.Sc. degree in electronics engineering and telecommunications from the Transilvania University of Braşov, in 2011 and 2013, respectively, and the Ph.D. degree, in 2017. Her Ph.D. dissertation was on Contributions to the System-on-Chip (SoC) Datasecurity Using PUF Circuits. She is currently a Teaching Assistant with the Department of Electronics and Computers, Transilvania University of Braşov. Her research is focused on SoC, multi-processor SoC, and security and cryptography. She was a recipient of the Romanian Patriot Fest Award for a security solution on SoC.



FLORIN D. SANDU was born in Braşov, Romania, in 1959. He received the Dipl. Eng. degree from the Electronics and Telecommunications Faculty of Bucharest, Romania, in 1984, and the Ph.D. degree from the Electrical Engineering and Computer Science Faculty, Transilvania University of Braşov, in 1998. His Ph.D. dissertation was on Data Acquisition Optimized for DSP. He was with the helicopter industry. Since 2001, he has been a Consultant for Siemens Romania, managing the technology transfer toward the university and a set of joint EU-funded research common projects. He is currently a Professor with Transilvania University, a Ph.D. Coordinator, and an In Charge of the Telecommunications Study Program. His research interests include telecommunications, telematics, data acquisition systems, virtual instrumentation, service science-management engineering, digital signal processing, applied electronics, and quality of electrical energy. He is with the Romanian Society for Electromagnetic Compatibility.