6

Securing the Internet of Things – Security and Privacy in a Hyperconnected World

Elias Z. Tragos[1], Henrich C. Pöhls[2], Ralf C. Staudemeyer[2], Daniel Slamanig[3], Adam Kapovits[4], Santiago Suppan[5], Alexandros Fragkiadakis[1], Gianmarco Baldini[6], Ricardo Neisse[6], Peter Langendörfer[7], Zoya Dyka[7] and Christian Wittke[7]

[1]FORTH, Greece
[2]University of Passau, Germany
[3]Technical University of Graz, Austria
[4]Eurescom, Germany
[5]Siemens AG, Germany
[6]European Commission, Joint Research Centre (JRC), Italy
[7]IHP, Germany

6.1 Introduction

The Internet of Things (IoT) introduces itself as a basic set of technological enablers to support the provision of innovative applications that can improve the quality of life of people and industrial productivity. IoT is increasingly supported by various stakeholders and market players that see clear business opportunities in this field. Cities have also identified the potential of IoT both for providing smart services to their citizens and for boosting the local economy by providing opportunities for new jobs and new businesses. Industry is considering IoT's adoption to drive Industry 4.0. All these are key reasons why IoT has attracted so much attention lately in both the research and the industrial world.

Main research areas in the IoT world until now included the development of technologies to efficiently interconnect large numbers of devices. Mobile phones and "dumb" devices (sensors and actuators) are being increasingly equipped with intelligence so that they are becoming able to act autonomously

for supporting new and advanced applications for healthcare, transportation, business control, and security, as well as energy and environmental monitoring.

Several estimations have been made for the number of devices that will be interconnected in the next few years and it looks like that billions of devices will be connected to the global Internet by 2020. In such a hyperconnected world, where all these devices are continuously monitoring their environment including the activities and everyday lives of citizens new threats arise regarding security and privacy. Providing a holistic security framework for IoT systems is not an easy task to do, because it requires cross-layer mechanisms [1] and systems needs to be designed to be secure and privacy preserving. Retrofitting security mechanisms in non-secure IoT systems can provide only a very limited level of security.

The aim of this chapter is to discuss the challenges for security and privacy in a hyperconnected world where humans are assisted by machines and technology, but not watched by or through them. Starting from the need to adopt the essence of "security and privacy by design", we discuss why there is a need to embed security mechanisms in a system from the conceptual phase through the design process to ensure a maximum level of data protection and to guarantee end-to-end security.

Firstly, we put the focus on the devices discussing two different research areas: (i) physical IoT security, namely what are the threats to IoT when someone has access to the physical device and how can we protect them and (ii) embedded security and privacy on the constrained devices and why a system cannot be fully secure without securing the devices that generate the data first. This latter part discusses several techniques, e.g. for lightweight encryption, data minimization, integrity protection and usage of gateways to enforce security policies close to the constrained devices.

Secondly, this chapter discusses the importance of protecting not only the data, but the metadata as well to ensure that the communication stays unobservable, providing also countermeasures regarding how to be protected from network traffic analysis. Access control based on trust policies is also an important research area in the IoT and is briefly discussed next, aiming to show the importance of context information in the decisions regarding access control.

Finally, we conclude with a discussion on enforcing security and privacy in the "Cloud", as more and more IoT systems are utilizing the cloud both for storage and processing of the IoT data. What type of security and (mainly) privacy mechanisms need to be applied in the cloud to protect the data is

currently an area which only lately started to receive attention and so far has not been properly explored, so we try to provide here an overview and suggest a way forward.

6.2 End-to-End Security and Privacy by Design

End-to-end security is a term that has quite distinct meanings depending on the OSI layer it refers to. In a hyperconnected IoT world, a multitude of networks and heterogeneous systems are bridged including a wide range of middleware systems that are all gathering, storing, and processing data. Then, from these huge amounts of data, information has to be generated to extract context for making smart decisions. Hence, end-to-end security between the devices and the applications is of paramount importance for protecting the privacy of people's personal data across the different systems and technologies that are involved. This requires strong data protection not limited to transit over wireless and intermediate Internet links, but also in all intermediate storage and processing points, till the data finally reaches solely its intended recipient.

The amount of acquired and processed data that will be ubiquitously provided in IoT becomes a huge concern for the people who are directly or indirectly monitored through their physical surroundings. Collection of personal information, starting even from their own devices and the surroundings they interact with, is high in quantity, quality and sensitivity. All this motivates the need for privacy in IoT [2].

The ubiquitous data collection in IoT is massive, even higher in comparison to other intrusive systems, such as online social networks and search engines [3]. While these generally trade privacy for commodity, their data collection depends on user interaction.

The ubiquity and pervasiveness of sensors to measure the status and context of an environment bring new types of privacy threats for the persons acting in that environment, regardless of them being users of the system or not. Thus, protecting the privacy of system participants as well as casual users and non-involved subjects in a future IoT is one of the main challenges for privacy-related research.

With the extensive data collection in mind it is clear that much of the business value lies in offering services that process and analyse the huge amounts of data collected [2, 4]. Nevertheless, these services should be as well privacy-enhanced, respecting and protecting the privacy of people's personal information. As a prerequisite for this the IoT systems must be built based

on the concept of "privacy by design", which means that privacy enhancing mechanisms must be deeply rooted inside the IoT architecture. Furthermore, the solution should be such that every data subject should be able to give consent to the collection, storage and processing of their personal data for the particular known in advance purpose (consent and purpose). These are the challenges of "Privacy by Design" in IoT, see [5, 6].

Tackling these challenges is one of the most important business factors of the future IoT. As stated in the Opinion 8/2014 of the Article 29 Data Protection Working Party: "Organisations which place privacy and data protection at the forefront of product development will be well placed to ensure that their goods and services respect the principles of privacy by design and are equipped with the privacy friendly defaults expected by EU citizens."

6.3 Physical IoT Security

The major concern when implementing cryptographic functions on constrained devices is efficiency, due to the fact that devices are battery driven and shall be working for years. Unfortunately, this focus may lead to a vulnerable network even though cryptographic functions may be supported by those devices. The issue here is that implementations of cryptographic algorithms may be insecure even if the algorithm is considered to be secure. An implementation may indirectly provide information on the keys used for example by its timing or energy consumption. This is especially dangerous in the IoT context since here at least for some applications we need to consider that devices can be stolen, analysed in a well-equipped lab and brought back. Due to the wireless communication and the fact that the devices can be unreachable for some time such attacks might go fully undetected. Next, we provide examples of such attacks and their countermeasures.

6.3.1 Selected Low-Cost Attacks

The strength of cryptographic algorithms according to the definition of Kerckhoff [7] is based only on the used key that is kept secret. This means a potential attacker may know the algorithm itself, plain text, encrypted text and even the length of the key. From this point of view cryptographic approaches are secure if the time for brute forcing is long, that is if length of the key is sufficient.

The main assumption here is that the cryptographic device is a black box for an attacker, assuming he knows the cryptographic function but cannot

get any details about how it is calculated. But in the IoT environments this assumption does no longer hold, due to the fact that devices may be stolen.

Even simple measurements like the ones of the current flowing through the chip or its electromagnetic radiation while a cryptographic function is calculated provide sufficient details to extract the key successfully. Such attacks – denoted as Side Channel Analysis (SCA) attacks – are often low-cost, easy and powerful. Even a single measurement can be sufficient to extract the cryptographic key in a few minutes for algorithms that are considered to be mathematically secure.

Figure 6.1 and Figure 6.2 show the same part of a measured PTs corresponding to processing the first 15 bits of the same cryptographic key using the same input on two different accelerators. It's a power trace of the elliptic curve point multiplication denoted as kP.

The calculations executed by two different IHP hardware accelerators of the kP operation of the standardized B-233 curve [8]. The shape of the measured traces is influenced by the private key, i.e. the shape of PTs while processing a '1' key bit differs from the shape of a '0'.

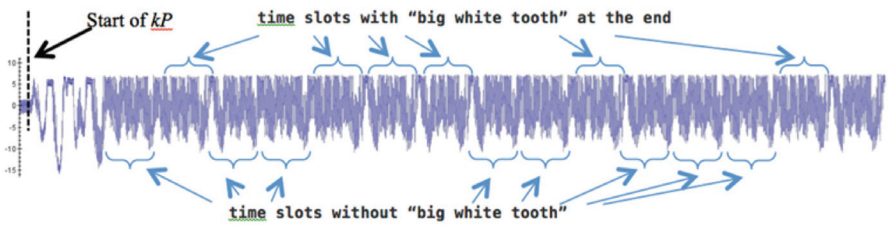

Figure 6.1 Implementation of cryptographic function – here the elliptic curve point multiplication – without paying attention to the SCA i.e. the shape of PT depends on the contents of the processed bit. This allows extracting the cryptographic key directly from the measured trace.

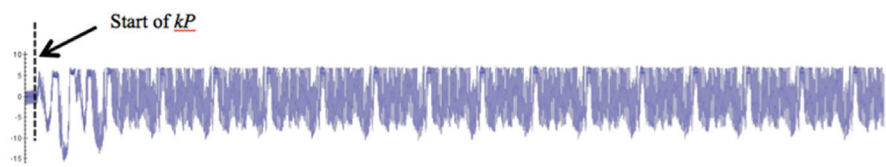

Figure 6.2 Implementation of the same cryptographic function taking SCA into account: the shape of the PT is always the same always, i.e. different key bits can no longer be distinguished in the PT. The cryptographic key cannot be extracted directly from measured trace.

If the cryptographic function is implemented without considering SCA this influence can be strong and the attacker can directly extract the key from a measured PT.

For example in Figure 6.1 two different kinds of the shapes are observable: time slots that have a "big white tooth" at its end and those without it. Using the assumption that the big white tooth at the end of the timeslots corresponds to the processing of a '1' key bit and other kind of slots corresponds to '0' key bit the used key can be correctly extracted.

Cryptographic algorithms implemented without considering SCA attacks can be called "weak" implementations. Knowledge about the details of such attacks – their main assumptions and the exploited characteristics – can help to implement the cryptographic algorithms in a way to be more resistant to such known attacks.

Figure shows the same part of the power trace of the same kP operation but executed on an improved version of our hardware accelerator, now all key bits are processed in the same way, i.e. simple power analysis attacks do not succeed.

Differential power analysis attacks are more powerful using statistical methods for analysis of measured traces. A very efficient, low-cost, fast and relatively easy attack on an elliptic curve cryptography (ECC) implementation is the horizontal power analysis using a difference of means test.

Each time slot can be observed as an independent curve. The mean curve of all slots can be calculated. After this the mean curve can be compared point wise with each slot.

If the power value of the mean curve is higher than the value of the current slot, it was assumed, this slot corresponds to the key bit value '1', otherwise to '0'. Thus, the first key candidate was obtained. Repeat this for all other points of the mean curve and you obtain the remaining key candidates.

We performed a horizontal power analysis attack using the difference of means test as described above for two simulated power traces. The power consumption of the IHP ECC design while processing the given EC point P using two different 232 bit long keys – k1 and k2 – was simulated using Synopsis Tools PrimeTime [9].

It obtained 57 key candidates for each of the investigated keys and we calculated the correctness of the extraction for each key. From a security point of view the ideal case is if the correctness of the key extraction is 50% for all key candidates. The green line in Figure 6.3 corresponds to this case.

Figure 6.3 demonstrates how powerful a difference of means based attack can be. In the case investigated here 225 bits of the 232 bit long 1st key

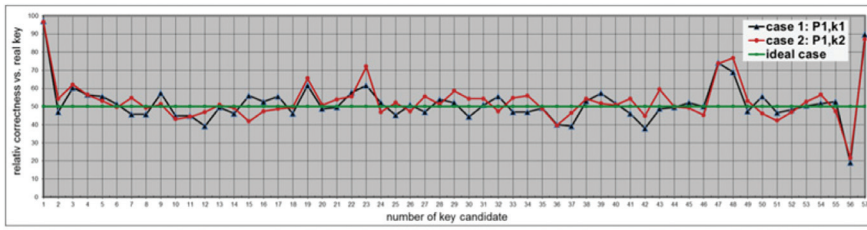

Figure 6.3 Relative correctness of the extraction of the key for each of the key candidates as a curve.

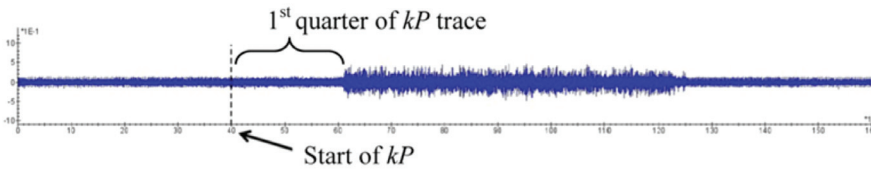

Figure 6.4 Difference of the traces of kP and of key candidate · P. The first quarter of the trace is equivalent to the noise, i.e. the quarter of the most significant bits of the key_candidate is the same as the quarter of the most significant bits of k while all remaining bits differ.

candidate were extracted correctly, i.e. the correctness is about 97% in both keys. The correctness of the next probable 4 key candidates is also high from 70% up to 90%.

The next power analysis attack (Figure 6.4) that we performed based on direct comparison of two traces is similar to the one first introduced in [10]. The main assumption here is that an attacker can run the device with a key candidate.

The idea is that the difference of two power traces is close to zero, i.e. is comparable to the noise, if the kP operation with the same EC point and with the same scalar k is performed. This means the key can be extracted serially, bit by bit.

Using the attack sketched above only about 100 measurements without any statistical processing of the measured data are necessary to extract a 232 bit long key k correctly. So the mathematically strong secure 232 bits long cryptographic key can be extracted correctly in a few hours only.

6.3.2 Key Extraction Attacks and Countermeasures

Figure 6.5 represents all types of attacks and countermeasures for public key cryptography. The diagram reflects that most of the attacks and countermeasures are based on a never expressed assumption i.e. the fact that the

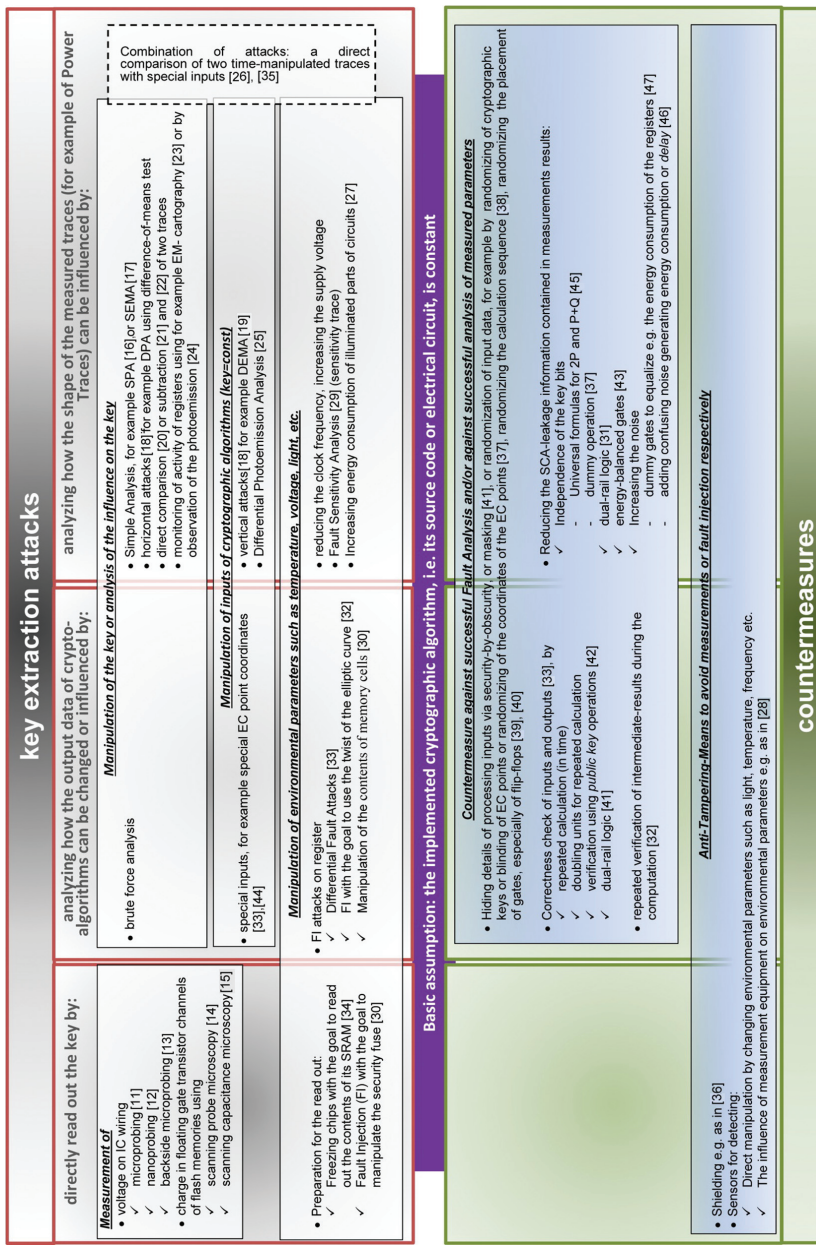

Figure 6.5 Key extraction attacks and countermeasures.

implementation of the cryptographic function is constant i.e. it cannot change during the attacks. This assumption is displayed as a rectangle connecting the attack and the countermeasure part.

The columns in the figure display:

- attacks in which the attacker can read out the key directly (left most column).
- changes in the output data (middle column).
- changes in the measured traces (right most column).

The rows represent all parameters that can be manipulated by the attacker i.e. the key candidate, other input data as well as the environmental parameter are given as rows.

The part of the Figure 6.5 representing the countermeasures is similarly structured as the attack part. The columns represent the same type of attacks as in the upper part of the diagram. The rows show countermeasures that (i) reduce the information that is contained in measurement results or that avoid access to faulty intermediate results, and (ii) avoid attacks by detecting the attack before it has any effect on the cryptographic implementation.

All countermeasures displayed in Figure 6.5 can help avoid attacks or at least hamper the potential success. Thus, they provide reasonable means to increase the security of the IoT. They come with some cost with respect to area and/or energy. Since cost of the devices and/or energy efficiency are paramount in IoT applications the use of countermeasures needs to be considered carefully. But in case physical access to IoT devices cannot be avoided and security is essential countermeasures need to be included in the implementations in order to ensure security.

6.4 On Device Security and Privacy

Designing a secure IoT system requires the embedding of security and privacy enhancing mechanisms locally on the devices near the physical entity of interest, whenever possible. Of course, this is much harder and more costly to maintain, e.g. it requires doing software updates for each smart device, which in turn requires reprogramming the actual device over the air, restarting it without needing human intervention and without configuring it again. Hardware must be capable of supporting advanced or even basic security mechanisms, as an insecure or non-private system design is hard to be turned later on into a privacy preserving or secure system.

The RERUM project tackles this with an "On-Device First" approach. RERUM's devices are made capable to run algorithms that enable the protection of security and privacy locally, by supporting advanced on-device security and privacy preserving mechanisms and over the air updating of the on-device software, while maintaining their energy consumption at very low levels.

6.4.1 Mediated Device Access for Security and Privacy

Security and privacy threats are continuously becoming more intelligent and they require more sophisticated countermeasures than IoT devices are capable of. Hence, we need an IoT gateway or IoT router to shield it. This is known as mediated device access. This gateway enables to hot-fix or firewall a large number of IoT devices from emerging threats, without the need to exchange every hardware device. Of course, if the local hardware device's privacy and security capabilities are outdated, the local threat level increases regardless of a gateway firewalling them from global threats. Thus, if one wants to secure the hotel building's management from the attacking hotel guest, each local device's security must be kept up-to-date.

Additionally to security, a gateway could be the local point of control and enforcement for privacy, as it has far more processing capabilities and gathers far more information from the environment than a single device. We assume that to apply privacy enhancing technologies (PET) the gateway would be trusted to act in the data subject's interest. Moreover, the gateway can use the diverse information it has from fusing other data from the data subject's devices as some form of ground truth or guidance, e.g., apply the PET differently when the data subject is at home or not. Mediated access to the lower end IoT devices, and hence some IoT gateway, is a necessity to ensure security and privacy.

6.4.2 Encryption

IoT mainly consists of severely resource constrained devices that are not capable of running complex encryption mechanisms like standard PCs. Thus, lightweight encryption mechanisms are of paramount importance for increasing the security of IoT. Lightweight cryptography normally provides adequate security but does not always consider energy efficiency. Symmetric key cryptography using Advanced Encryption Standard (AES) [48] is widely used in practical implementation of encryption based on block ciphers on constrained devices. Hash functions (e.g. SHA-3 [49]) are also widely used

but they are not lightweight, and only lately there are some research steps towards lightweight hash functions. Elliptic Curve Cryptography (ECC) [50] is used in IoT due to the fact that it uses keys of much smaller size than standard public key cryptography mechanisms. However, its execution time might still not be fast enough for some devices.

The majority of existing encryption algorithms do not fully fulfil the requirements for energy efficiency. Furthermore, key distribution schemes are necessary for their proper operation, making the network vulnerable to adversaries that manage to capture the keys during key exchange. Basic requirements for efficient lightweight IoT encryption can be assumed to be the following:

- Encryption mechanisms have to be optimized for their energy efficiency. This is critical as sensors are resource constrained devices in terms of memory, CPU, and processing;
- Key distribution schemes should be avoided or their usage should be minimized. These consume valuable energy, and there is also the risk of information hijacking (by an adversary) during the key exchange and
- Keys should not be pre-stored on the sensor device (currently this is done usually during manufacturing). This poses a significant security threat as sensors can be easily compromised when placed in outdoor environments.

In Wireless Sensor Networks (WSNs) the Compressive Sensing (CS) technique has been widely used for compressing the data that are gathered by sensors. CS is a very useful technique because it applies at the same step both data compression and lightweight lossy encryption [51]. The reconstruction error is directly related with the level of compression and encryption and the nature of the signal that is captured by the sensor. For example, a slow varying temperature signal has very low reconstruction error, while another signal that has rapid changes will result to a very high reconstruction error.

Within RERUM, a technique for extracting the encryption keys for CS at real-time has been proposed, supporting the requirement for not hardware-coding the keys on the IoT devices [52]. Key extraction is performed using channel measurements, thus there is no need for any key distribution mechanism. The derived keys are used for encryption/decryption using the primitives of CS. Evaluation results have shown that legitimate nodes experience a very low reconstruction (decryption) error, while adversaries located at a distance greater than half of the carrier frequency's wavelength, experience a higher error, thus being unable to capture and decode sensitive information.

6.4.3 Integrity

Integrity is the "property that data has not been altered [. . .] in an unauthorised manner"[1]. In a hyperconnected world, the IoT's flow of communication is highly loosely coupled, meaning that data that are transmitted over a secure channel are then stored and processed or transmitted further later. Protecting the integrity for those type of loosely connected data can be achieved by message-level protection mechanisms. Using a cryptographically secure signature scheme, based on asymmetric keys, allows verifying that data has not been modified in unauthorised ways. Additionally, you gain origin-authentication, i.e., verifying which entities' public key signed the data. Adding a message authentication code (MAC), with a shared key between sender and receiver, also allows ensuring that the message's integrity cannot be violated without being detected by the receiver.

6.4.4 Data Minimisation

In [53], the authors underline that the very foundation of privacy by design is data minimization, which is defined as the property to limit as much as possible the release of personal data and, for those released, preserve as much unlinkability as possible [54]. To exemplify how data minimization is related to privacy by design, the reader is referred to the popular Privacy By Design framework [55].

If personal data collection is minimized from the very beginning, much less effort will be needed to further define and implement appropriate privacy enhancing mechanisms. The application of adequate technologies for data minimization requires expertise in the services that the IoT system provides. The engineer must decide if it is possible to render the same (or comparable) functionality with less amount of personal information. In some cases, unlinkability might not always be desirable, for instance if devices and data must be needed to be linked to a user, for billing, authentication or otherwise.

The best place to achieve data minimization is on the devices where the data are sensed, as the amount of personal information can be minimized before the data are transmitted from the devices to the backbone system. This can be enforced with hard privacy mechanisms, such as malleable signatures and group signatures [56], which can be implemented on devices to ensure integrity and create unlinkability for data. Location privacy technologies [57] can be applied on devices e.g. to measure traffic data and compute averages

[1] ETSI TS 133 105 V10.0.0 (2011-04)

of speed and distance, while anonymizing a participant's real location in the geolocation system.

In addition to privacy preserving technologies for sensed data, further privacy mechanisms are needed for quasi-identifiers such as metadata and IP-addresses can provide sensitive information. Traffic analysis, as one example, is frequently used to identify the sources of data and thus de-anonymize the information. Mechanisms to ensure communication observability can further enhance privacy protection for the IoT, which are discussed in the following section.

6.5 Unobservable Communication

Even if the protection of user data is addressed by means of end-to-end encryption in the future, we still need to look into information loss caused by leaking protocol metadata. This leakage can go up to the point, which may render end-to-end encryption obsolete. To reduce it, at least the following properties [58] shall be preserved by the network of IoT devices:

- Coding – All messages with the same encoding can be traced.
- Size – Messages with the same size can be correlated.
- Timing – By observing the duration of a communication and considering average round-trip times between the communication partners patterns of network participation can be extracted.
- Counting – The number of messages exchanged between the communicating parties can be observed.
- Volume – Volume combines information gained from message size and count. The volume of data transmitted can be observed.
- Pattern – By observing communication activity, patterns of sending and receiving can be observed.

Furthermore, message frequencies and flow can be analysed. The message flow between parties includes both the traffic volume and communication pattern. Communication partners have a unique distinguished behaviour that can be fingerprinted. An observer can perform a brute force analysis of the network by observing all possible paths of communication and generating a list of all possible recipients.

Finally the observer can also perform a long term intersection/disclosure analysis of the network by observing devices and the network for long time and reducing the set of possible communication paths and recipients by analysing online and offline periods. Characteristic usage patterns, such as an IoT device

connecting every minute, may appear and can be used to further reduce the number of possible paths.

The following Table summarises the message properties and how they can be addressed.

Table 6.1 Message properties

Attacks Based on	Proposed Solutions
Message Coding	Change coding during transmission e.g. with k-nested encryption
Message Timing	1) batched forwarding of messages 2) random delay of messages ($\text{delay}_{\min} \geq \text{latency}_{\max}$)
Message Size	Use a predefined message size and padding small messages
Message Counting	Receive and forward a standard number of messages and use dummy traffic
Communication Volume	Protect message size and communication volume
Communication Pattern	Continuous network participation
Message Frequency	Use a standardized message exchange pattern
Brute Force	No clear protection dummy traffic helps
Long Term Intersection	No clear protection continuous connectivity and dummy traffic help

6.5.1 Resisting Network Traffic Analysis

Leakage of metadata can be reduced by providing protection against network traffic analysis. This includes endpoints, timing and location information. Traffic analysis can be addressed by ensuring unobservable communication as implemented by anonymising networks using generally proxy chains. Anonymising proxy networks have started with the implementation of Chaum's Mix in 1981 [59]. The system tunnels encrypted traffic through a number of low-latency proxies, as depicted in Figure 6.6.

Initially, interest in this field was primarily theoretical but in the last 30 years a lot of research in this field has looked at developing practical and usable systems for preserving anonymity [60, 61]. Such systems cover Email, Web browsing and other services like peer-to-peer networks and IRC chat. Systems like The Onion Router (TOR) and the Invisible Internet Project (I2P) allow generic layer 3 transmission. While TOR was primarily developed to

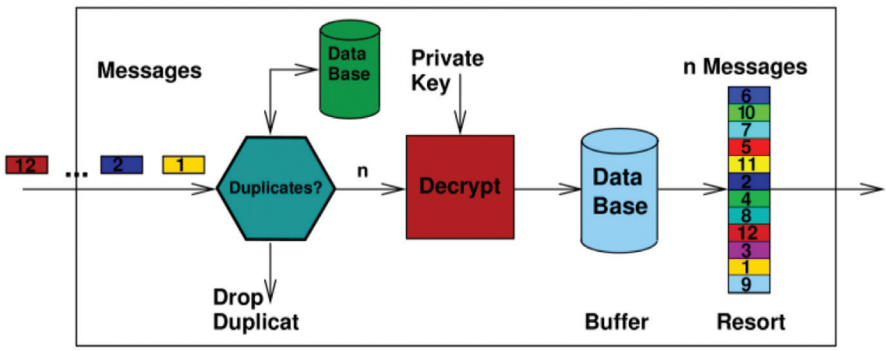

Figure 6.6 Chaum's MIX.

allow anonymous web browsing in close to real-time the general concept is applicable to prevent traffic analysis in the IoT network.

Once traffic leaves the TOR network it can be observed, therefore end-to-end encryption is needed and is the responsibility of the end nodes. Apart from TOR, there is I2P, an anonymous/pseudonymous network layer. Like TOR, I2P can be extended for many services. I2P is neither as secure nor as fast as TOR, but can handle large volumes of traffic, like those foreseen for the IoT.

6.6 Access Control Based on Policy Management

A policy management framework (as developed in the iCORE project) supports IoT-specific access control requirements like the hyperconnected-ness and distributed-ness of the IoT and the need of applications to share resources and even data. The Security Toolkit (SecKit) [62] models the IoT system for security specification purposes. The system design is divided into an entity domain and a behaviour domain, with an assignment relationship between entities and behaviours.

In the entity domain, the entities and the communication mechanisms allowing the entities to exchange information are specified. In the behaviour domain the behaviour of each entity is detailed including actions, interactions, causality relations, and information attributes. It is also possible to specify the data, identity, context, trust, role, risk, and security rules in so called metamodels. Figure 6.7 illustrates their dependencies.

The context metamodel specifies Context Information and Context Situation types. Context Information is a simple type of information about an entity that is acquired at a particular moment in time, and Context Situations are a

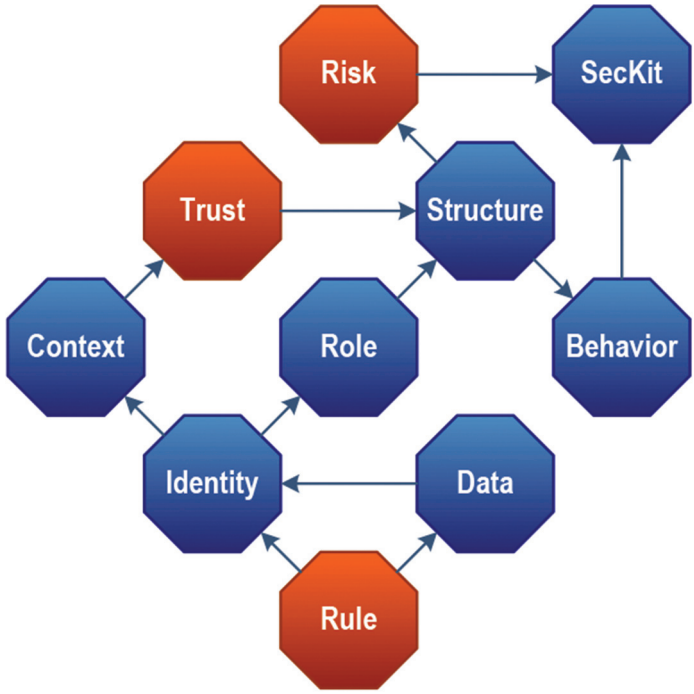

Figure 6.7 SecKit metamodels and dependencies.

complex type that models a specific condition that begins and finishes at specific moments in time [63]. For example, the "Body Temperature" is a Context Information type, while "Fever" is a situation where a target patient has a temperature above 37 degrees Celsius. Entities are associated to context situations using roles (e.g. patient).

A Context Manager component monitors and registers events when situations begin and end. These events contain references to the entities that participate in the situation and can be used to support the specification of the policy rules. Policy rules can be specified to represent authorizations to be granted when a situation begins and data protection obligations that should be fulfilled when the situation ends. For example, access to the patient data can be allowed when an emergency situation starts with the obligation that all data is deleted when the emergency ends. A security policy may be specified to allow access to data when the situation starts and to trigger the deletion of the data when the situation ends.

The security policies have to be disseminated to the device that is gathering the data under consideration in a secure way. Depending on the security policy, the device has to trigger and apply the appropriate mechanism for transmitting the data in the exact format needed by the application. This includes a two-step process; (i) at first the device has to map the policies for the application to specific data gathering policies and (ii) then it should identify the encryption/security level of the data to identify the proper transmission mechanisms, considering also the energy efficiency requirements of the devices (using i.e. an adaptive encryption scheme). For example, in a traffic monitoring scenario, users in cars may be sending information regarding traffic to an application, which should know only how much traffic there is at every street segment. The users' phone has the ability to send various types of traffic related data, i.e. exact location every second, speed every second, direction of movement, etc. If the application wants to estimate the traffic, the related policies should be considered by the devices of the users, so only an average speed per time period and street segment is sent, in order to avoid disclosing the exact location of the user at each point of time (ensuring privacy by design). Actually, intermediate nodes (i.e. the gateway) should also consider these policies and send to the application server only aggregated/average data so that the location of the users will be hidden from the application point of view. Other applications that need to know the exact location of the user (depending on their access control policies) will indeed be identified as such by the devices, which will transmit the exact location (i.e. for a person to track his car if it is stolen). It is evident, thus, that the transmission of the security policies to the devices is of crucial importance for ensuring the security and privacy of the overall system. The system should be able to identify the integrity of the policies that are sent to the devices, so that unauthorized applications will not gain access to privacy-sensitive data.

The security rules model supports the specification of rule templates (a.k.a. policies) to be enforced and the configuration rules to instantiate these templates. Templates can be specified considering the security and privacy non-functional requirements of confidentiality, data protection, integrity, authorization, and non-repudiation. The security rule templates are Event-Condition-Action rules, with the Action part being an enforcement action of Allowing, Denying, Modifying, or Delaying an activity carried out by an IoT device or application. Furthermore, the Action part may also trigger the execution of additional actions to be enforced, or to specify trust management policies to increase/decrease the trust evidence for a specific trust aspect.

From a trust management perspective, the SecKit supports the specification of aspect-specific trust relationships and exchange of trust recommendations. For example, trust relationships can be defined for identity provisioning aspect, privacy protection, data provisioning, and so on. A trust relationship also includes a trust degree, which is mapped to a Subjective Logic (SL) opinion considering the amount of belief, disbelief, and uncertainty [64]. The aspect-specific approach considering uncertainty is more realistic from a human perspective since people usually trust others for specific purposes (e.g., a mechanic to fix your car) and most of the time cannot be absolutely certain about the amount of trust they may place.

The security policy rules can be delegated from one administrative domain to another when the domains interact and exchange data. For example, when a smart home exchanges data with a smart vehicle, the smart home can exchange the policies that regulate the authorizations and obligations associated to the exchanged data that should be enforced by the smart vehicle. This delegation of sticky flow policies must be supported by trust management mechanisms [63] in order to guarantee or increase the level of assurance with respect to the enforcement of the policy rules by the smart vehicle.

6.7 Security and Privacy in the IoT Cloud

The "Cloud" complements quite well the IoT supporting the storage and processing of the large amounts of data that are gathered by constrained devices. However, the Cloud introduces new threats for security, but especially with respect to information privacy. When IoT data are moved to the cloud for storage we could use encryption to protect it. However, if the application turning that data into information is running in the cloud, then the cloud provider becomes yet another third party that needs to process the stored data gathered from the physical world. Hence, the provider inherits all the privacy problems of the data. In fact, the third party becomes part of the IoT application provider's own computation and storage infrastructure. However, the cloud provider is technically not under its full control. This situation has shown to be problematic and incidents recently showed that economic incentives and legal tools used to increase trust in the service provider, e.g. Service Level Agreements, are by far not sufficient to guard personal data and trade secrets against illegal interceptions, insider threats, or vulnerabilities exposing data in the cloud to unauthorized parties. While being processed by a cloud provider, data are typically neither adequately protected against

unauthorized read access, nor against unwanted modification, or loss of authenticity. Consequently, in the most prominent cloud deployment model today – the public cloud – the cloud service provider (CSP) necessarily needs to be trusted. Next, we will provide some selected areas from PRISMACLOUD's cryptographic research and highlight their foreseen suitability for IoT data.

6.7.1 Verifiable and Authenticity Preserving Data Processing

Verifiable computing allows checking the result of a computation for its validity, even if the computation itself was done by one or more untrusted processing units. While respective systems have already been implemented, they have not yet seen real-world deployment. Besides general purpose systems [65] there are various approaches that are optimized for specific (limited) classes of computations or particular settings [66]. A cloud user can facilitate those mechanisms to check if collected measurements have been processed correctly, and, if not so, they can identify (maliciously) incorrect calculations.

When data are subject to computations executed by the cloud provider, it is extremely helpful if the processing allows preserving the authenticity of data that are manipulated by computations. The most generic tool for preserving authenticity under admissible modifications are (fully) homomorphic signatures (or message authentication codes) [67]. Signatures with more restricted capabilities, like redactable signatures introduced in [68, 69], offer a restricted set of capabilities, but with better performance [70, 71]. Redactable and sanitizable signatures have been proven to strongly preserve the privacy [72] of the original values when they have been updated/changed or redacted [73]. They allow preserving the authenticity on data introduced by an IoT device's signature, which vouches for the data's origin, even after processing. Thus, the cloud user after authorized processing can still verify the involved data's authenticity.

6.7.2 Structural Integrity and Certification of Virtualized Infrastructure

Structural integrity and certification of virtualized infrastructures connects attestation of component integrity, i.e., proving the trustworthiness of claims about the infrastructure, and security assurance of cloud topologies, i.e., guaranteeing that a cloud topology provides certain security guarantees (e.g., network isolation). This is a clear benefit for cloud infrastructure consumers as their confidence in infrastructure properties can be increased and the cloud

provider can be held accountable. The recent concept of graph signatures [74] is a promising candidate to connect the two aforementioned areas. They allow a trusted third-party auditor to digitally sign a graph and prove in zero-knowledge properties of the graph, such as connectivity or isolation. Graph signatures can be a handy tool for a cloud provider to convince a certain customer in a multi-tenant environment that the granted infrastructure fulfils certain security properties, but at the same time to not disclose the confidential blueprint of the virtualized infrastructure. For instance, the cloud may prove that a customer's part of the infrastructure is isolated from other tenants without revealing how their part of the infrastructure looks like. A first implementation of a system that allows certification of and proofs about a certified infrastructure as well as other relevant and interesting use-cases has already been outlined in [75]. This allows to attest an IoT's infrastructure in a way to ensure that certain security properties are satisfied and improves accountability.

6.7.3 Privacy Preserving Service Usage and Data Handling

Privacy-preserving service usage essentially means to realize 1) data minimisation, i.e., to only reveal information that is essential for service delivery, and 2) avoid (behavioural) tracking of service users. This is especially important in cloud based applications, as such information may, among others, reveal confidential business information [76]. Attribute-based anonymous credential (ABC) systems and related concepts such as group signature schemes [77] are important concepts for realizing such privacy-preserving applications. They allow users to authenticate in an anonymous way, i.e., without revealing their identity, but allow to prove claims that enable a service provider to still make access decisions. Although they are quite mature in the research community, they still lack practical adoption, which, however, needs to be considered as a very important topic for future cloud IoT applications.

Another issue is privacy in context of data handling. In contrast to achieving data privacy by means of encryption, which realizes an all-or-nothing mechanisms for the access to the data, we thereby mean scenarios which are often encounter when processing of data by third parties in the cloud is required. Essentially, this covers mechanisms for data anonymisation such that a provable level of anonymity can be achieved, i.e., k-anonymity [78] or differential privacy [79]. In particular, one requires a guarantee that when (large amounts of) structured data are given away or are dynamically queried, it can be ensured that a targeted degree of privacy is guaranteed, i.e.,

data collected from many individuals does not allow to uniquely identify single individuals but still allows to compute meaningful statistical parameters. Techniques for privacy-preserving service usage allow IoT devices to anonymously authenticate to services and prevent linking of transactions conducted by IoT devices. Data anonymisation can help to protect privacy of individuals if IoT devices send sensitive information (e.g., health data) to the cloud and the data is later released for further processing.

6.7.4 Confidentiality of (Un-)structured Data

Confidentiality of data when outsourced to the cloud for the purpose of storage and/or processing is considered to be sine qua non, since cloud providers can neither be considered as fully trustworthy nor are resistant to attacks. Encryption is a classical tool to provide confidentiality. Unfortunately, encryption clearly limits the functionality (how to operate on data), adding encryption to legacy applications may cause serious problems and the management of the involved cryptographic keys soon becomes highly complex. Within the last years, significant research has been put into cloud storage solutions that distribute the data to multiple clouds (aka cloud-of-cloud approach) [80]. They allow providing confidentiality for data at rest with strong security in a key-less manner under some non-collusion assumption and thus solve the key management problem (at least partially). An interesting challenge is to design such a distributed architecture using active nodes to fully delegate secure multi-user storage to the cloud. Thereby, the use of efficient Byzantine protocols helps to improve robustness and various types of secret sharing protocols can help to cope with different adversary settings. Furthermore, for a multi-user setting a trustworthy distributed access control mechanism is required and it is interesting to extend it with access privacy features. Another issue, as mentioned above, is the integration of encryption into legacy (e.g., database) applications, as they may be unable to use or store encrypted data, causing them to crash or alternatively, to output incorrect values. Techniques like format-preserving encryption (FPE) [81], order-preserving encryption (OPE) [82] and tokenization schemes have emerged as very useful tools as they can be directly applied without adapting the application itself.

6.7.5 Long Term Security and Everlasting Privacy

Classical cryptographic primitives such as digital signature schemes and encryption schemes are valuable tools to achieve integrity, authenticity, and confidentiality. If these properties, however, need to hold in the long-term,

e.g., for some decades or even indefinitely, these tools often fail. Cryptanalytic progress and advances in computing power can reduce their security or may even make them entirely worthless. There are only few approaches that consider long-term confidentiality, integrity and authenticity. Moreover, many of the existing solutions lack in providing these properties [83, 84].

6.7.6 Conclusion

At the moment privacy guarantees with respect to user's IoT gathered data in the cloud can only be given on a contractual basis and rest to a considerable extent on organizational (besides technical) precautions. Companies or individuals alike are in the end a cloud user, and they themselves are responsible for their data's privacy, whether processing gets outsourced to the cloud or not.

Therefore, the H2020 project PRISMACLOUD is looking into novel security and privacy preserving methods, such that cloud usage can be facilitated even by organizations that deal with highly sensitive data such as health data and maintaining security critical services. PRISMACLOUD only just started in the first quarter of 2015. But the vision is that only a new generation of cryptographically secured cloud services with security and privacy built in by design can lead the way to achieving the required privacy properties for outsourced data storage and processing at the upper end of the IoT – privacy in the cloud.

6.8 Outlook

Security and privacy in the IoT world are research areas that only lately have attracted the attention of both the research and the industrial world. Up until now, the focus was limited on creating efficient middleware platforms to enable the services to gather data from the devices. This resulted in existing IoT deployments that are not secure and gather all types of personal information. Fortunately, recently the significant focus on security and privacy has resulted in important achievements not only in the technology domain, but also on the way the world sees the IoT. Security and privacy are now basically seen as the key points for the wider adoption of the IoT applications by the general public. If the citizens can be reassured that the IoT will not harm them, will not steal their private information and will not affect their lives in a negative way, only then they will gladly accept and embrace IoT and the full potential of IoT can unfold to improve their – and everyone's – quality of life.

This chapter presented a cross-layer approach on improving the security and the privacy of IoT systems, allowing them to work for the benefit of the

people, without leaking information, presenting a risk or damaging people's privacy. Designing a system so complex as the IoT whilst guaranteeing that a certain level of security is achieved is an extremely complex and tedious task, and can not be retrofitted, so we must already design the IoT with privacy and security in mind. It is widely acknowledged that the security of an IoT system depends heavily on the devices, so we need to physically secure the IoT devices, as every system's level of security is as good/high as the one of its weakest part.

Although encryption can really contribute to protecting the data that are being exchanged in an IoT system, this is not quite enough. Even with encryption deployed end-to-end, the IoT still leaks information by communication metadata. If the volume and quality of the information collected is sufficiently large, even encrypted information can be extracted without breaking the encryption of the communication channel.

From what was previously described, it is also quite important to design the system to be privacy preserving, starting from embedding in the devices mechanisms for both data minimization and for enhancing privacy. These are quite important to ensure that the services will only get the exact data they need and nothing more, to avoid the possibility of linking data.

But we need to think even broader, the problem of privacy – and of security – well extends into the cloud. The society has to be able to trust the whole IoT value chain all the way up to the cloud. Thus, new cryptographically proven security and privacy mechanisms must be developed to allow provably using cloud services securely and privately.

In general, there is a lot of work done in the IoT world towards enhancing the security and the privacy of IoT systems. However, making significant progress in this area through research is not enough. The industrial world and the businesses need to put more focus on embracing and adopting security and privacy solutions. To complete the picture, regulations for protecting IoT data need to be put into place, to ensure the adherence of every player to the socially accepted norms of privacy in the EU. Only then the hyperconnected world of the IoT becomes not a threat to the citizens, but a useful tool to improve not only our's – but everyone's – everyday lives.

Acknowledgment

This work is partially funded by the EU FP7 projects RERUM (GA no 609094), SMARTIE (GA no 609062), iCore (GA no 287708) and the H2020 project PRISMACLOUD (GA no 644962).

Bibliography

[1] Henrich C. Pohls, et al. RERUM: Building a reliable IoT upon privacy- and security-enabled smart objects. *Wireless Communications and Networking Conference Workshops (WCNCW),* IEEE, 2014.

[2] Jacob Kohnstamm, Drudeisha Madhub: Mauritius Declaration on the Internet of Things. *In: 36th International Conference of Data Protection and Privacy Comissioners*, 2014.

[3] Ralph Gross, and Alessandro Acquisti. Information revelation and privacy in online social networks. *Proceedings of the 2005 ACM workshop on Privacy in the electronic society. ACM,* 2005.

[4] Elgar Fleisch: What is the Internet of Things? An Economic Perspective. *In: Economics, Management, and Financial Markets 2* (2010), S. 125–157.

[5] Rodrigo Roman, Jianying Zhou, Javier Lopez: On the features and challenges of security and privacy in distributed internet of things. *In: Computer Networks 57 Nr. 10, 2266–2279. Towards a Science of Cyber Security Security and Identity Architecture for the Future Internet,* 2013.

[6] Marc Langheinrich: Privacy by Design — Principles of Privacy-Aware Ubiquitous Systems. Version: 2001. http://dx.doi.org/10.1007/3-540-45427-6_23. In: ABOWD, GregoryD. (Hrsg.); BRUMITT, Barry (Hrsg.); SHAFER, Steven (Hrsg.): Ubicomp 2001: *Ubiquitous Computing Bd.* 2201. Springer Berlin Heidelberg, 2001. – ISBN 978-3-540-42614-1, 273–291.

[7] D. Hankerson, A. Menezes, S. Vanstone: *Guide to Elliptic Curve Cryptography*. Springer-Verlag New York, Inc. (2004).

[8] National Institute of Standards and Technology: Digital Signature Standard (DSS), FIPS PUB 186-4, July 2013, http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

[9] Synopsis – PrimeTime, http://www.synopsys.com/Tools/Implementation/SignOff/Pages/PrimeTime.aspx

[10] T. S. Messerges, E. A. Dabbish, R. H. Sloan: Power Analysis Attacks of Modular Exponentiation in Smartcards. *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems* – CHES 1999, LNCS Volume 1717, pp. 144–157, Springer Berlin Heidelberg, 1999.

[11] Z. Dyka, P. Langendörfer: Improving the Security of Wireless Sensor Networks by Protecting the Sensor Nodes against Side Channel Attacks. *Wireless Networks and Security, Signals and Communication Technology* 2013, pp. 303–328, Springer Berlin Heidelberg, 2013.

[12] Hitachi: Nano-Probing System, http://www.hitachi-hitec.com/global/em/nan/nan_index.html

[13] C. Helfmeier, D. Nedospasov, C. Tarnovsky, J. S. Krissler, C. Boit, J.-P. Seifert: Breaking and Entering through the Silicon. *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security* – CCS 2013, November 4–8, 2013, Berlin, Germany, pp. 733–744.

[14] Ch. De Nardi, R. Desplats, Ph. Perdu, F. Beaudoin, J. L. Gauffier: EEPROM Failure Analysis Methodology: Can Programmed Charges Be Measured Directly by Electrical Techniques of Scanning Probe Microscopy? *Proceedings of the 31st International Symposium for Testing and Failure Analysis* – ISTFA 2005, November 6–10, 2005, San Jose, CA, USA.

[15] Ch. De Nardi, R. Desplats, Ph. Perdu, Ch. Guérin, J. L. Gauffier, Th. B. Amundsen: Direct Measurements of Charge in Floating Gate Transistor Channels of Flash Memories Using Scanning Capacitance Microscopy. *Proceedings of the 32nd International Symposium for Testing and Failure Analysis* – ISTFA 2006, November 12–16, 2006, Austin, Texas, USA.

[16] S. A. Kadir, A. Sasongko: Simple power analysis attack against elliptic curve cryptography processor on FPGA implementation, *Proceedings of International Conference on Electrical Engineering and Informatics,* pp. 1–4 , 17–19 July 2011, IEEE.

[17] E. De Mulder, P. Buysschaert, S. B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, I. Verbauwhede: Electromagnetic Analysis Attack on an FPGA Implementation of an Elliptic Curve Cryptosystem. *EUROCON 2005—International Conference on Computer as a Tool,* November 21–24, 2005, Belgrade, Serbia and Montenegro, pp. 1879–1882.

[18] C. Clavier, B. Feix, G. Gagnerot, M. Roussellet, V. Verneuil: Horizontal Correlation Analysis on Exponentiation. *Proceedings of the 12th International Conference on Information and Communications Security – ICICS 2010*, December 15–17, 2010, Barcelona, Spain, LNCS Volume 6476, pp. 46–61, Springer Berlin Heidelberg, 2010.

[19] E. De Mulder, S. B. Ors, B. Preneel, I. Verbauwhede: Differential Electromagnetic Attack on an FPGA Implementation of Elliptic Curve Cryptosystems. *World Automation Congress – WAC*, July 24–26, 2006, Budapest, Hungary.

[20] M. Hutter, M. Kirschbaum, T. Plos, J. M. Schmidt, S. Mangard: Exploiting the Difference of Side-Channel Leakages. *Proceedings of the Third International Workshop on Constructive Side-Channel Analysis and Secure Design – COSADE 2012*, May 3–4, 2012, Darmstadt, Germany, LNCS Volume 7275, pp. 1–16, Springer Berlin Heidelberg, 2012.

[21] Z. Dyka, Th. Basmer, Ch. Wittke, P. Langendoerfer: *Individualizing Electrical Circuits of Cryptographic Devices as a Means to Hinder Tampering Attacks*, Cryptology ePrint Archive 2015/442.

[22] T. S. Messerges, E. A. Dabbish, R. H. Sloan: Power Analysis Attacks of Modular Exponentiation in Smartcards. *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems – CHES 1999*, LNCS Volume 1717, pp. 144–157, Springer Berlin Heidelberg, 1999.

[23] J. Heyszl, S. Mangard, B. Heinz, F. Stumpf, G. Sigl: Localized Electromagnetic Analysis of Cryptographic Implementations. *Proceedings of the Cryptographers' Track at the RSA Conference – CT-RSA 2012*, San Francisco, CA, USA, February 27–March 2, 2012, LNCS Volume 7178, pp. 231–244, Springer Berlin Heidelberg, 2012.

[24] A. Schlösser, D. Nedospasov, J. Krämer, S. Orlic, J. P. Seifert: Simple Photonic Emission Analysis of AES. *Proceeding of Cryptographic Hardware and Embedded Systems – CHES 2012*, LNCS Volume 7428, pp. 41–57, Springer Berlin Heidelberg, 2012.

[25] J. Krämer, D. Nedospasov, A. Schlösser, J.-P. Seifert: Differential Photonic Emission Analysis. *Constructive Side-Channel Analysis and Secure Design – COSADE 2013*, LNCS Volume 7864, 2013, pp. 1–16, Springer Berlin Heidelberg, 2013.

[26] P.-A. Fouque, F. Valette: The Doubling Attack – Why Upwards Is Better than Downwards. *Cryptographic Hardware and Embedded Systems – CHES 2003*, LNCS Volume 2779, pp. 269–280, Springer Berlin Heidelberg, 2003.

[27] S. Skorobogatov: Optically Enhanced Position-Locked Power Analysis. Proceedings of 8th International Workshop *Cryptographic Hardware and Embedded Systems – CHES 2006*, Yokohama, Japan, October 10–13, 2006, LNCS Volume 4249, Springer Berlin Heidelberg, 2006.

[28] Cl. Helfmeier, Chr. Boit, U. Kerst: On Charge Sensors for FIB Attack Detection. *Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust – HOST 2012*, San Francisco, CA, USA, Jun. 2012, pp. 128–133.

[29] H. Sakamoto, Y. Li, K. Ohta, K. Sakiyama: Fault Sensitivity Analysis against Elliptic Curve Cryptosystem. *Proceedings of the 2011 Workshop*

on Fault Diagnosis and Tolerance in Cryptography – FDTC 2011, pp. 11–20.

[30] S. P. Skorobogatov: *Semi-invasive attacks – a new approach to hardware security analysis.* Technical Report UCAM-CL-TR-630, University of Cambridge, Computer Laboratory, April 2005.

[31] S. Moore, R. Anderson, P. Cunningham, R. Mullins, G. Taylor: Improving Smart Card Security using Self-timed Circuits, *Proceedings of the International Symposium on Advanced Research in Asynchronous Circuits and Systems* – ASYNC 2002, pp. 211–218.

[32] P. Fouque, R. Lercier, D. Real, F. Valette: Fault Attack on Elliptic Curve with Montgomery Ladder Implementation. *Proceedings of the 5th Workshop on Fault Diagnosis and Tolerance in Cryptography – FDTC 2008*, August 10, 2008, Washington, DC, USA, pp. 92–98.

[33] I. Biehl, B. Meyer, V. Müller: Differential Fault Attacks on Elliptic Curve Cryptosystems. *Proceedings of the 20th Annual International Cryptology Conference – CRYPTO 2000*, Santa Barbara, CA, USA, August 20–24, 2000, pp. 131–146, Springer Berlin Heidelberg, 2000.

[34] S. Skorobogatov: *Low temperature data remanence in static RAM*. Technical Report UCAM-CL-TR-536, University of Cambridge, Computer Laboratory, June 2002.

[35] N. Homma, Atsushi Miyamoto, Takafumi Aoki, Akashi Satoh, Adi Shamir: Collision-based Power Analysis of Modular Exponentiation Using Chosen-message. *Proceedings of the 10th International Workshop – CHES 2008*, LNCS Volume 5154, pp. 15–29, Springer Berlin Heidelberg, 2008.

[36] S. Briais, J.-M. Cioranesco, J.-L. Danger, S. Guilley, D. Nacchache, Th. Porteboeuf: Random Active Shield. *Proceedings of the Workshop on Fault Diagnosis and Tolerance in Cryptography – FDTC 2012*, September 9, 2012, Leuven, Belgium, pp. 103–113.

[37] J. Coron: Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems. *Proceedings of the First International Workshop – CHES 1999*, August 12–13, 1999, Worcester, MA, USA, LNCS Volume 1717, pp. 292–302, Springer Berlin Heidelberg, 1999.

[38] E. Oswald, M. Aigner: Randomized Addition-Subtraction Chains as a Countermeasure against Power Attacks. *Proceedings of the Third International Workshop – CHES 2001*, May 14–16, 2001, Paris, France, LNCS Volume 2162, pp. 39–50, Springer Berlin Heidelberg, 2001.

[39] K. Itoh, T. Izu, and M. Takenaka: A Practical Countermeasure against Address-Bit Differential Power Analysis. *Cryptographic Hardware and*

Embedded Systems – CHES 2003, Proceedings of the 5th International Workshop, Cologne, Germany, September 8–10, 2003, LNCS Volume 2779, pp. 382–396, Springer Berlin Heidelberg, 2003.

[40] J. Heyszl: *Impact of Localized Electromagnetic Field Measurements on Implementations of Asymmetric Cryptography*. Technische Universität München, Lehrstuel fuer Sicherheit in der Informationstechnik an der Fakultaet fuer Elektrotechnik und Informationstechnik, Dissertation, 2013.

[41] J.-S. Coron, L. Goubin: On Boolean and Arithmetic Masking against Differential Power Analysis. *Proceedings of the Second International Workshop – CHES 2000*, Worcester, MA, USA, August 17–18, 2000, LNCS Volume 1965, pp. 231–237, Springer Berlin Heidelberg, 2000.

[42] A. Shamir US 5991415.

[43] K. Tiri, M. Akmal, I. Verbauwhede: A Dynamic and Differential CMOS Logic with Signal Independent Power Consumption to Withstand Differential Power Analysis on Smart Cards. *Proceedings of the 28th European Solid-State Circuits Conference – ESSCIRC 2002*, September 24–26, 2002, pp. 403–406.

[44] L. Goubin: A Refined Power-Analysis Attack on Elliptic Curve Cryptosystems. *Proceedings of the 6th International Workshop on Practice and Theory in Public Key Cryptography – PKC 2003*, Miami, FL, USA, January 6–8, 2003, LNCS Volume 2567, pp. 199–211, Springer Berlin Heidelberg, 2002.

[45] E. Brier, M. Joye: Weierstraß Elliptic Curves and Side-Channel Attacks. *Proceedings of the 5th International Workshop on Practice and Theory in Pubic Key Cryptosystems – PKC 2002*, Paris, France, February 12–14, 2002, LNCS Volume 2274, pp. 335–345, Springer Berlin Heidelberg, 2002.

[46] J.-S. Coron, I. Kizhvatov: An Efficient Method for Random Delay Generation in Embedded Software. *Proceedings of the 11th International Workshop – CHES 2009*, Lausanne, Switzerland, September 6–9, 2009, pp. 156–170, Springer Berlin Heidelberg, 2009.

[47] J. Goodman, A. P. Chandrakasan: An Energy Efficient Reconfigurable Public-Key Cryptography Processor Architecture. *Proceedings of the Second International Workshop – CHES 2000*, Worcester, MA, USA, August 17–18, 2000, LNCS Volume 1965, pp. 175–190, Springer Berlin Heidelberg, 2000.

[48] J. Daemen, V. Rijmen, *The Design of Rijndael: AES – The Advanced Encryption Standard*. Springer, 2002. ISBN 3-540-42580-2.

[49] NIST Computer Security Division (CSD). *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions* (PDF). NIST.

[50] D. Hankerson, A. Menezes, and S.A. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer-Verlag, 2004.

[51] E. Candes and M. Wakin, *An introduction to compressive sampling*, IEEE Signal Processing Magazine, vol. 25, no. 2, pp. 21–30, 2008.

[52] A. Fragkiadakis, E. Tragos, and A. Traganitis. Lightweight and secure encryption using channel measurements. *Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), 2014 4th International Conference on*. IEEE, 2014.

[53] S. Gürses, C. Troncoso, and C. Diaz. *Engineering privacy by design*. Computers, Privacy & Data Protection 14 (2011).

[54] A. Pfitzmann, and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. (2010): 34.

[55] A. Cavoukian, Evolving FIPPs: Proactive Approaches to Privacy, Not Privacy Paternalism. *Reforming European Data Protection Law*. Springer Netherlands, 2015. 293–309.

[56] D. Chaum, and E. Van Heyst. Group signatures. *Advances in Cryptology—EUROCRYPT'91*. Springer Berlin Heidelberg, 1991.

[57] H. Tschofenig, et al. The IETF Geopriv and presence architecture focusing on location privacy. *Position paper at W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*, Ispra, Italy. 2006.

[58] J. Raymond, *Traffic analysis: Protocols, attacks, design issues, and open problems*. Designing Privacy Enhancing Technologies, 10–29, 2001.

[59] D. L. Chaum, Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2), 84–90, 1981.

[60] A. Ruiz-Martínez, A survey on solutions and main free tools for privacy enhancing Web communications. *Journal of Network and Computer Applications*, 35(5), 1473–1492, 2012.

[61] G. Danezis, & R. Clayton, Introducing traffic analysis. *In Digital Privacy: Theory, Technologies, and Practices* (pp. 1–24), 2007.

[62] R. Neisse, I. Nai Fovino, G. Baldini, et al. A Model-based Security Toolkit for the Internet of Things. *International Conference on Availability, Reliability and Security (ARES)*, University of Fribourg, Switzerland, 2014.

[63] R. Neisse, D. Holling, A. Pretschner, Implementing Trust in Cloud Infrastructures. *11th IEEE/ACM International Symposium on Cluster,*

Cloud and Grid Computing (CCGRID), Newport Beach, USA, May 2011.

[64] A. Jøsang, Evidential reasoning with subjective logic, *in: 13th International Conference on Information Fusion*, 2010.

[65] M. Walfish, A. J. Blumberg,: Verifying Computations without Reexecuting Them. Commun. ACM 58(2), 74–84 (2015).

[66] M. Backes, D. Fiore, R. M. Reischuk, Verifiable delegation of computation on outsourced data. In: *ACM CCS*. pp. 863–874. ACM (2013).

[67] D. Catalano, Homomorphic Signatures and Message Authentication Codes. *In: SCN. LNCS*, vol. 8642, pp. 514–519. Springer (2014).

[68] R. Johnson, D. Molnar, D. Song, D. Wagner, Homomorphic Signature Schemes. In: *CT-RSA*. pp. 244–262. LNCS, Springer (2002).

[69] R. Steinfeld, L. Bull, Content Extraction Signatures. In: *ICISC*. Springer (2002).

[70] H. C. Pöhls, K. Samelin, On updatable redactable signatures. In: *ACNS*. pp. 457–475. LNCS, Springer (2014).

[71] C. Brzuska, H. C. Pöhls and K. Samelin. Efficient and Perfectly Unlinkable Sanitizable Signatures without Group Signatures. In *Proc. of the 10th European Workshop: Public Key Infrastructures, Services and Applications (EuroPKI 2013)*, pages 12–30, Springer Berlin Heidelberg, 2013.

[72] H. C. Pöhls and K. Samelin. On Updatable Redactable Signatures. *In Proc. of the 12th International Conference on Applied Cryptography and Network Security (ACNS 2014)*, Springer, 2014.

[73] C. Brzuska, H. C. Pöhls and K. Samelin. Efficient and Perfectly Unlinkable Sanitizable Signatures without Group Signatures. *In Proc. of the 10th European Workshop: Public Key Infrastructures, Services and Applications (EuroPKI 2013)*, pages 12–30, Springer Berlin Heidelberg, 2013.

[74] T. Groß, Certification and efficient proofs of committed topology graphs. In: *CCSW*. ACM (2014).

[75] T. Groß, Signatures and Efficient Proofs on Committed Graphs and NP-Statements. *In: Financial Cryptography and Data Security. LNCS*, Springer (2015).

[76] Y. Chen, V. Paxson, R. H. Katz, What's New About Cloud Computing Security? University of California, Berkeley, Tech. Rep. UCB/EECS-2010-5.

[77] J. Camenisch, A. Lehmann, G. Neven, Electronic Identities Need Private Credentials. IEEE Security & Privacy 10(1): 80–83 (2012).
[78] P. Samarati, k-Anonymity. Encyclopedia of Cryptography and Security (2nd Ed.) 2011: 663–666.
[79] C. Dwork, Differential Privacy. Encyclopedia of Cryptography and Security (2nd Ed.) 2011: 338–340.
[80] D. Slamanig, C. Hanser,. On cloud storage and the cloud of clouds approach. *ICITST 2012*, IEEE (2012).
[81] J. Black, P. Rogaway, Ciphers with Arbitrary Finite Domains. *CT-RSA 2002*, LNCS, Springer (2002).
[82] R. Agrawal, J. Kiernan, R. Srikant, Y. Xu, Order-Preserving Encryption for Numeric Data. *SIGMOD Conference 2004*, ACM (2004).
[83] J. Braun, J. A. Buchmann, C. Mullan, A. Wiesmaier, Long term confidentiality: a survey. Des. Codes Cryptography 71(3): 459–478 (2014).
[84] V. Gagliotti, M. A., Buchmann, J. A., Cabarcas, D., Weinert, C., Wiesmaier, A. Integrity, authenticity, non-repudiation, and proof of existence for long-term archiving: A survey. Computers & Security 50: 16–32 (2015).