MDPI

Article

A Study on Security and Privacy Guidelines, Countermeasures, threats: IoT Data at Rest Perspective

Hezam Akram Abdulghani, Niels Alexander Nijdam, Anastasija Collen, Dimitri Konstantas

Geneva School of Economics and Management, Geneva University, Switzerland;
{mohammed.akram | niels.nijdam | anastasija.collen | dimitri.konstantas}@unige.ch
* Correspondence: mohammed.akram

Version June 24, 2019 submitted to Symmetry

Abstract: Internet of Things (IoT) makes our lives much easier, more valuable, and less stressful due to the development of so many applications around us like smart city, smart car, and smart grid that offer endless services and solutions. Protecting IoT data at rest either on the objects or in the cloud of such applications is an indispensable requirement for achieving a symmetry in the handling and protection of IoT, as we do with data created by persons and applications. This is because unauthorised access to such data may lead to harmful consequences such as linkage attacks, loss of privacy, and data manipulation. Such undesired implications may jeopardise the existence of IoT applications, if protection measures are not taken, and they stem from two main factors. One is IoT objects have limited capabilities in terms of memory capacity, battery life, and computational power that hamper the direct implementation of conventional Internet security solutions without some modifications like traditional symmetric algorithms. Another factor is the absence of widely-accepted IoT security and privacy guidelines for IoT data at rest and their appropriate countermeasures, which would help IoT stakeholders like developers and manufacturers to develop secure IoT systems and, therefore, enhance IoT security and privacy by design. To this end, we first briefly describe the main IoT security goals and identify IoT stakeholders. Moreover, we briefly discuss the most well-known data protection frameworks such as GDPR and HIPAA. Second, we highlight potential attacks and threats against data at rest and show their violated security goals (e.g., confidentiality and integrity). Third, we review a list of protection measures by which our proposed guidelines can be accomplished. Fourth, we propose a framework of security and privacy guidelines for IoT data at rest that can be utilised to enhance IoT security and privacy by design and establish a symmetry with the protection of user created data. Our framework also presents the link between the suggested guidelines, mitigation techniques, and attacks. Moreover, we state those IoT stakeholders like manufacturers and developers who will benefit most of these guidelines. Finally, we suggest several open issues required further investigation in the future, and we also discuss the limitations of our suggested framework.

Keywords: Internet of Things; security guidelines; privacy guidelines; countermeasures; security goals; attacks, IoT data at rest

1. Introduction

The Internet of Things (IoT) is a network of objects equipped with sensors, actuators, electronics and connectivity protocols enabling object interaction with each other without human intervention [1]. IoT is involved in the creation of a variety of applications and services around us, for example in smart cities, smart cars, smart grids, quality of life applications, and electronic gadgets, all of which make our lives more productive and less stressful. For instance, the authors in [2] have proposed an IoT system

which can be used to manage students' stress. It is unquestionable that such IoT applications and services provide a huge benefit for human's life, yet they may come with a massive cost for individual's privacy and security protection. This is because IoT inherits most of the issues of Internet associated with location awareness, security, quality of service [3] and, most likely, amplifies them due to direct connection with physical objects [4,5].

As IoT applications may store their data locally on objects or remotely in the cloud, based on their storage capabilities, protecting their data at rest is of paramount importance. Several IoT applications may cooperate with each other to accomplish specific tasks or services. If data integrity of a single IoT application at rest has been compromised, then there is a huge risk of dealing with a cascading effect of the data compromise. For instance, the authors in [6] state that a thermostat deployed in a smart home relies heavily on a smoke detector's data to shut a heating system down in case of danger. However, accessing the smoke detector's data by unauthorised objects may put the entire smart home at risk. The countermeasure on ensuring the data integrity across multiple Cloud Storage Services (CSSs) was proposed by [7], where reliance on the Third-Part Auditor (TPA) for data verification was eliminated by use of a decentralised blockchain based Integrity Management Service. Furthermore, once an IoT system stores its data in the cloud, there is no assurance that only authorised objects or users will have access to their data. An example was given by European Union Agency for Network and Information Security (ENISA)[1], where an employee at the SharpLocks company, attacker, was capable (due to given access rights) to send a malicious update from the company's server to all connected IoT objects [8].

However, most security and privacy issues of IoT data at rest, such as unauthorised access and weak or absent encryption schemes, arise from two principal reasons. IoT objects have limited capabilities in terms of computational power, memory, and bandwidth [9]. Because of these limitations, a direct implementation of traditional security mechanisms into IoT objects tends to be very hard without some modifications. This is why a new breed of lightweight IoT security techniques and protocols (e.g., a secure system of uploading and replicating IoT data suggested in [10]) has been developed [11,12]. The second reason, which motivates us to conduct this work, is the lack of widely-accepted security and privacy guidelines for IoT at data at rest, along with their appropriate mitigation techniques. The main objective of such guidelines and countermeasures is to improve IoT security and privacy by design by giving IoT stakeholders, such as manufacturers and developers, a chance to embrace such guidelines and countermeasures from the early stages of IoT system development[13]. If IoT stakeholders overlook these guidelines when dealing with IoT data at rest, the appearance of several attacks and threats is inevitable.

It is clear that unauthorised access and privacy violations of individuals, associated with data at rest, will appear again and again in IoT systems, unless the mindset of all IoT stakeholders shifts to properly integrate at early stages the security and privacy guidelines for IoT data at rest, along with their corresponding countermeasures. The first step towards this paradigm shift is the development of a comprehensive set of security and privacy guidelines. However, there is a complete lack of research efforts conducted specifically to such objective. To the best of authors' knowledge, there is no paper or document clearly addressing security and privacy guidelines for IoT data at rest along with their mitigation strategies. The existing research proposals [14–21] focus primarily on IoT guidelines in general. They neither provide a comprehensive guidelines for IoT data at rest, nor discuss their proper countermeasures. A detailed explanation of such efforts along with a brief comparison between our suggested guidelines for IoT data at rest and theirs is presented in Section 2.

The main contributions of this work are the following:

1. To highlight IoT security goals as well as IoT stakeholders.
2. To summarise the attacks and threats against IoT data at rest.

[1] https://www.enisa.europa.eu/

3. To review a set of implementation techniques by which our suggested guidelines can be implemented and also states those IoT stakeholders who will benefit from these guidelines.
4. To propose a framework of security and privacy guidelines for IoT data at rest that can be utilised to reinforce IoT security and privacy by design.
5. To discuss open issues, limitations and future work.

The rest of the article is structured as follows. In section 2, we present the current research studies on IoT guidelines with focus on data at rest, describe the most popular data protection frameworks and highlight the IoT security goals and distinguish IoT stakeholders. Section 3 discusses threats and attacks on IoT data at rest. We identify the appropriate techniques for mitigating the identified attacks on IoT data at rest in Section 4. A proposed framework on security and privacy guidelines is presented in Section 5. Finally, we discuss open issues for further investigation for future work in Section 6.

2. Related work

In this section, we outline the current state of the art related to IoT security guidelines, existing frameworks on data protection and identify the involved stakeholders specific for the IoT environment. We limit our discussion to the guidelines on protecting IoT data at rest, as the main topic of this work.

2.1. Research efforts on IoT guidelines

In [14], the authors have proposed a set of security and privacy guidelines for IoT data at rest, such as minimise data storage, minimise data retention, encrypt data storage, and time-period data aggregation. Furthermore,attacks and threats against IoT data at rest are analysed. Having said that, the authors do not provide a comprehensive set of guidelines for IoT data at rest, nor do they state the required implementation techniques to achieve their guidelines.

In [15], the BITAG suggests a list of security and privacy guidelines for IoT data at rest like minimise data storage, encrypt data storage, remove the sensitive data, and ensure data availability. BITAG, however, does not offer a thorough set of guidelines for IoT data at rest, nor does it identify the countermeasures required to carry out its guidelines. Furthermore, threats and attacks against IoT data at rest remain unchecked.

In [16], the Open Web Application Security (OWASP) suggests different security and privacy guidelines for IoT data at rest, such as minimise data storage, minimise data retention, ensure authorised access, and prevent physical access. In addition, the OWASP states those IoT stakeholders like manufacturer, developer, and customer who may use its guidelines to protect IoT data at rest. Nevertheless, the OWASP neither recognised the required countermeasures to implement its guidelines, nor distinguishes possible attacks and threats against IoT data at rest.

In [17], the ENISA proposes several security and privacy guidelines for IoT data at rest, such as minimise data retention, encrypt data storage, define recovery strategies, inform customers, and proper data destruction. The ENISA, however, does not recognise those IoT stakeholders who may utilise its guidelines, nor does it identify proper solutions to apply its guidelines. Moreover, the ENISA uncovers attacks and threats against IoT data at rest.

In [18], the IoTA suggests a list of security and privacy guidelines for IoT data at rest, example of which are encrypt data storage, inform customers, and remove sensitive data. That said, IoTA neither distinguishes the required implementation techniques to fulfil its guidelines, nor points out IoT stakeholders who may use its guidelines. Attacks and threats against IoT data at rest also are left unidentified.

In [19], IoT Security Foundation (IoTSF) proposes a set of security and privacy guidelines for IoT data at rest, such as distribute data storage, define recovery strategies, proper data destruction, and search on encrypted data. IoTSF, however, neither recognised suitable countermeasures to realise its guidelines, nor distinguishes attacks and threats against IoT data at rest.

In [21], the authors proposes a comprehensive set of security and privacy guidelines for the first two levels of CISCO'S reference model (edge nodes and communication). Even though their

guidelines are not meant specifically for protecting IoT data at rest, only three of these guidelines (ensure authorised access, remove or hide sensitive data, and search on encrypted data) can be used to do so. It is worth mentioning that the authors identify all possible threats and attacks against edge nodes and communication, state those IoT stakeholders who may use their guidelines, and recognise suitable implementation techniques to implement them.

2.2. Data protection frameworks

The lack of efficient standards, regulation, and weak governance is a cause of security and privacy issues of IoT. However, some initiatives at a national level have been proposed, which we briefly present in the next paragraphs.

General Data Protection Regulation (GDPR): In December of 2016, European Union (EU) has voted to use GDPR as a replacement of outdated Data Protection Directive (DPD) proposed in 1996. The main goal of DPD was to preserve personal information of individuals within EU from being misused and allow the individuals of EU to have better control over their personal data. The GDPR is intended to substitute the DPD as a regulation, and it will cover the whole EU as unified law. The GDPR has included six major changes (e.g., personal data redefined, individuals rights, data controller and processor, and global impact) compared to the DPD. The detailed explanation of each one of them, for interested readers, can be found in [22]. To preserve personal data, the GDPR has imposed six fundamental principles, the detail of which can be found in [23]. It is worth mentioning that not all IoT applications are dealing with personal information, for instance Industrial Internet of Things (IIoT) applications. However, this may be the case in the majority of IoT systems (e.g., healthcare). It is obvious that the market of IoT solutions is increasing worldwide including Europe and it is a relevant topic which needs thorough management. One example is a digital transformation of healthcare due to the fast growth of wearable and interconnected medical objects which provide a remote health monitoring. Therefore, healthcare data is highly sensitive data, and it attracts the attention of attackers. To this end, authors in [24] identified different attacks associated with IoT multi-cloud e-Healthcare environment, such as side-channel attacks and malicious insider attacks. Healthcare data, thus, must be covered under the scope of the GDPR. Other important IoT applications, which deal with personal data like smart metering and smart home applications, must also be covered under the scope of GDPR.

Health Insurance Portability (HIPAA): The main goal of HIPAA is to protect individuals from losing their health insurance if they have pre-existing health problems or they change their jobs. HIPAA, however, has been extended over the years to minimise the administrative and the cost burdens of healthcare processes. Most recently, HIPAA concentrates on developing standards as well as requirements to ensure security and privacy of Personal Health information (PHI) which can be created, stored, or transferred in several formats (e.g., written documents and verbal conversations). PHI may contain anything in patient health records like images, names, email addresses, and other information. As patients demand their data to be secure, HIPAA's security and privacy rules, discussed in details in [25], require healthcare organisations to embrace a set of processes and procedures to assure the highest level of patient confidentiality. Under HIPAA, a covered organisation may not utilise or reveal PHI unless it has received an explicit consensus from a patient to do so. It is unquestionable that IoT will change the healthcare experience. To this point, several examples are available on the market to illustrate how IoT has simplified the process of care management. For instance, collecting information in users' homes will assist healthcare providers to comprehend user's health in a comprehensive way, choose suitable treatment plans, alter the plans as time moves, and most importantly anticipate future health actions. It is clear that IoT can be utilised by healthcare organisations to reduce costs and at the same time enhance health outcomes in patients. Hence, healthcare IoT solutions must be covered under the scope of HIPAA.

Industrial Internet Consortium (ICC): It is an organisation developed in 2014 to improve the growth of interconnected objects. The primary goal of such organisation is to build an alliance of companies, academia , and governments and cooperate on the development of test beds for real-world systems. Furthermore, it has actively involved in supporting the necessities of standards in the IoT industry. To this end, ICC in 2016 has proposed a security framework developed specifically for IIoT. The main purpose of this security framework is to establish global industry acceptance on how to develop secure IoT systems [26].

IoTSF: It is a non-profit company, which has been reinforcing the IoT industry since 2015. This reinforcement includes developing security and privacy guidelines, courses, and training. IoTSF also has addressed the issues in the industry, and more importantly it has struggled to cover the gab via a cooperative initiative with many companies dealing with IoT. This kind of cooperation attempts to share expertise, knowledge, and enhancing best practices. To contribute to such objectives, IoTSF in [19], for example, has proposed an IoT framework as a checklist which can be used by the IoT manufacturers to simplify their compliance to IoTSF'framework.

2.3. IoT security goals and stakeholders

In this section, we first discuss security goals specific to IoT environment. Traditional security goals, in the literature, are broken down into three primary sets: (i) Confidentiality, (ii) Integrity and (iii) Availability, referred to as the CIA-triad. Confidentiality assures that only authorised objects or users can get access to sensitive data. As several IoT objects might deal with sensitive data, such as medical records and credit cards, the confidentiality of such data must be preserved. The impact of unauthorised access to medical objects, which may reveal personal data or lead to life-threatening situations, have been illustrated in [27]. In the IoT context, integrity is also crucial, since it ensures IoT data has not been tampered. If the integrity of IoT data has been compromised, undesired consequences may take place, for instance revealing a patient's privacy as a result of compromising his/her insulin pump [28]. IoT availability is essential, as it guarantees that IoT data is available and accessible to its users. Even though the Confidentiality, Integrity and Availability triad (CIA-triad) is popular, it fails to address new threats which appear in a cooperative environment, according to [29]. To tackle this issue, the authors in [29] propose a complete list of security goals called Information, Assurance, and Security octave (IAS) octave, known as the IAS octave, by studying a huge number of information in the state of the art in terms of security. Table 1 summarises the security goals suggested by the IAS octave, along with their definitions and abbreviations in link with IoT environment.

Table 1. IoT Security goals[4]

Security Requirements	Definition	Abbreviations
Confidentiality	Only authorised objects or users can get access to the data	CONF
Integrity	Data completeness and accuracy is preserved	INTG
Non-repudiation	IoT system can validate the occurrence of any event	NREP
Availability	Ensuring accessibility of an IoT system and its services	AVAL
Privacy	Presence of privacy rules or policies	PRIV
Auditability	Monitoring of the IoT object activity	AUDI
Accountability	End users can take charge of their actions	ACNT
Trustworthiness	Reliability on IoT object identity	TRST

In order to build the framework of security and privacy guidelines, suitable to all aspects of IoT environment life-cycle, we first propose a classification of identified IoT stakeholders into four groups, depicted in Table 2. It relates the main stakeholders with their associated role, in order to dictate the degree of guideline adaptation and stakeholder impact.

Table 2. IoT stakeholders [21]

Stakeholders	Roles	Abbreviations
Manufacturer	Building IoT hardware products	MAN
Developer	Developing IoT software solutions or services	DEV
Provider	Providing services for IoT products to customers	PRV
Consumer	Using IoT objects in different aspects of their daily lives	CNS

3. Attacks on IoT data at rest

In this section, we describe attacks and threats applicable for IoT data at rest and correlate them with IoT security goals, identified in Table 1. More specifically, we annotate with '⚠' when security goal in question is violated by the described attack.

***(AT1)* Misuse of data remnants:** This type of attack takes place when IoT objects are taken in possession by other end users either by voluntarily life cycle transitions (second hand, recycling) or due to loss, theft, malfunctioning and other involuntarily causes. Even though there is possibility that data of such objects may be deleted before selling them, it is not always the case for all sold objects. Those objects which store valuable information during their entire life cycle (e.g., personal photos and passwords) could be a primary target for many attackers, since they are not discarded properly during their end-of-life period or their data is not purged completely during attempted data cleanup [30]. This attack directly violates all security goals as the attacker has full control and access to the physical storage.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠	⚠	⚠	⚠	⚠	⚠	⚠

***(AT2)* Linkage attack:** The probability of unauthorised access, as well as leaks of sensitive information, grows significantly with the data sources of IoT systems linked, where each additional link creates exponential exposure. When intercepting and cross referencing between multiple data sources leads to partial data identification it is called linkage attack [31]. This attack violates *CONF*, *INTG* and *PRIV* security goals, as the attacker is manipulating the intercepted data without directly interfering with the actual IoT object(s). Therefore, other security goals are not applicable.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠			⚠			

***(AT3)* Data manipulation:** illegitimate modification of data at rest can be accomplished in two methods: (i) exploiting several vulnerabilities in application program interface such as SQL injection, and cross site scripting, and (ii) taking advantage of weak security mechanisms like small or weak passwords[32], [33]. Similarly to the "Misuse of data remnants" attack, all security goals are violated as the attacker operates directly on the IoT object.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠	⚠	⚠	⚠	⚠	⚠	⚠

***(AT4)* Side-channel attacks:** This attack is based on the discovery of information by analysing exposed side properties of the algorithmic implementation, such as processing timing, power consumption, or even associated sounds. This type of attacks may take place due to the absence of secure methods of processing and storing IoT data, for instance storing unencrypted data either in

the cloud or on IoT objects. The authors in [34] have discussed several data leakage attacks on CSSs, such as a confirmation of a file and learning the content of files. In the case of confirmation of a file, an adversary who already knows the plain text content of a file, can examine if a duplicate of the file has been stored elsewhere in the CSS. In the case of learning the contents of the files, the adversary can reveal highly sensitive data, since the attacker already recognises most the of file and attempts to guess or identify the unknown segments of file by examining whether the output of the encryption meets the observed cipher text. Similarly to the "Linkage attack", the *CONF*, *INTG* and *PRIV* security goals are violated as the attacker is indirectly revealing the private data, already generated and processed by the IoT object.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠			⚠			

***(AT5)* Denial of Service attacks:** Denial of Service (DoS) attacks in cloud computing prevent CSSs from offering their normal services or solutions for a period of time. The authors in [35] stated that DoS attacks which compromise the availability of such CSS stem from many contributing factors, the most notable of which are resource exhaustion, process disruption, physical disruption, and data corruption. For instance, the authors in [31] have stated that an attacker could flood a CSS with fake data at high frequency, which in turn makes such storage service spending most of its time validating the authenticity of such data and, therefore, is not able to timely reply to any valid requests. The inability of timely response may cause a delay which is not preferable for most IoT applications, specifically real-time applications like air traffic system and NFC Payment. For interested readers, the recently published survey of DoS attacks in the cloud can be found in [35]. First of all the *AVAL* is affected by this attack, as implied by the definition of the attack. *ACNT* is also no longer guaranteed due to the slow response times of the system. For *INTG* the guaranteed transmission and/or storage can be compromised, especially for real-time applications. *AUDI* is also violated, since the system can not perform continuous of the objects activity

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
	⚠		⚠		⚠	⚠	

***(AT6)* Insider attack:** An insider threat takes place when either a former or a current user, who has authorised access to an object's data or CSSs, misuses such access rights to compromise security goals of IoT, such as confidentiality, integrity, and availability. Malicious insiders can be considered as a considerable threat to many organisations. They could adversely influence a company's mission and reputation, and therefore they could pose a major impact to any business. The cyber security intelligence index [36] in 2016 has stated that 60 percent of all attacks are derived by the insiders. From this percentage, the malicious insiders had a great contribution, since the majority of such attacks (44.5 percent) was caused by them. EY in its recently published survey in [37] has identified several types of insider threats such as fraud, infrastructure sabotage, and unauthorised trading. In the context of IoT, the whole IoT ecosystem, starting from objects located in different environments and their data and applications in the cloud, may be vulnerable to insider threats. To this end, the authors in [24] have illustrated the applicability of malicious insider attacks in all the layers of IoT multi-cloud based e-healthcare architecture composed of 4 layers. In layer 1 (physical) where several sensors are deployed to collect the health data of different patients, an insider attacker, in this case, could alter the settings to send wrong data to the healthcare companies. Not only that, the attacker could obtain and reveal the patients' information. Likewise, in layer 2 (network) where many connectivity protocols (e.g., Bluetooth low energy) are utilised to transfer the patient data to the next layer, the malicious insider could carry out many unwanted activities like redirecting the packets to a vicious network

and compromising the availability of health data by initiating DoS attack on the network. In layer 3 (cloud), the malicious insiders could perform a set of malicious activities like gaining unauthorised access to patients' data, altering the e-health applications, modifying data stored in the storage, and executing collision attacks. Also layer 4 (application) is susceptible to malicious insider attacks. This is because any authorised entity from lower to upper level could uncover or alter the patients' heath data, which, for sure, will impact the level of trust among patients, doctors, and health organisations [24]. Similarly to the "Misuse of data remnants" attack, all security goals are violated as the attacker operates directly on the IoT object.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠	⚠	⚠	⚠	⚠	⚠	⚠

***(AT7)* Homogeneity attack:** As several anonymisation-based solutions have been proposed, such as k-anonymity and t-closeness techniques, some of them lack a method in which the diversity of their sensitive attributes is not supported, making such techniques like *k*-anonymity vulnerable to homogeneity attacks. This attack is applicable on the cases where there are identical records within data sets. Similarly to the "Linkage attack", the *CONF*, *INTG* and *PRIV* security goals are violated as the attacker is indirectly capable of attributing of the private data to specific identities.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠			⚠			

***(AT8)* Unauthorised access:** IoT data is vulnerable to different attacks because of storing it either in IoT objects or remotely in the cloud with no supervision of their holders. It is also expected that the number of threats and attacks will be intensified, since attackers can get access to such data once it is not properly protected due to the absence of strong encryption techniques. Furthermore, data might be placed in several data centres located at different countries, and such countries may have a high power to access such data without permission of their holders[38,39]. Another example of unauthorised access can be found in [40]. The authors stated that an adversary may get access to IoT data illegitimately during the migration procedure of virtual machine to a untrusted host which might reveal its sensitive data. Similarly to the "Misuse of data remnants" attack, all security goals are violated as the attacker has a direct access to the data on the IoT object or in the CSS.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠	⚠	⚠	⚠	⚠	⚠	⚠

***(AT9)* Identification:** An identification attack can be considered one of the most common threat against IoT data in which an attacker could link some identifier attributes (e.g., name and address) with some individuals. Similarly to the "Linkage attack", the *CONF*, *INTG* and *PRIV* security goals are violated as the attacker is indirectly capable of attributing of the private data to specific identities.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠			⚠			

***(AT10)* Hash collision:** The key goal of the collision attack is to reveal two input strings of a hash function that gives the same hash value. Due to the fact that a hash function has variable input lengths and a short fixed length output, there is a possibility that two different inputs generate the same output and this case is known as a collision[41,42]. As a consequence, an attacker can compromise

the encryption key and therefore intercept or have an access to the IoT object's data. Similarly to the "Linkage attack", the *CONF*, *INTG* and *PRIV* security goals are violated as the attacker is indirectly revealing the private data, already generated and processed by the IoT object.

CONF	INTG	NREP	AVAL	PRIV	AUDI	ACNT	TRST
⚠	⚠			⚠			

4. Mitigation techniques for protecting data at rest

In this section, we analyse existing methods on IoT data protection and attribute these mitigation techniques for the attack vectors, identified in Section 3.

***(MT1)* Deduplication schemes:** Attributed to attack *AT4* and *AT8*. Data deduplication is a method in which only a unique copy of redundant IoT data is stored, and links (not actual data) to the copies are provided. This is why such technique can be used as a backup strategy. Therefore, the development of secure deduplication schemes, capable of detecting identical data copies and storing them once, is a need and challenge at the same time.To this end, several data deduplication techniques in literature have been proposed which can be classified broadly into two categories (server-side and client-side) based on location at which data deduplication is accomplished[31].

It is worth noting that despite the benefits of deduplication schemes in saving disk space, minimising network bandwidth, and preventing unauthorised access, such techniques are susceptible to side-channel attacks. For instance, the authors in [34] have stated that implementing deduplication techniques in cloud storage may cause side-channel attacks like identifying files, learning the contents of files, and a covert channel. The authors also have illustrated several practical solutions like encryption and proof of ownership to mitigate such attacks. To this point, a few secure deduplication techniques have been proposed, described below:

In[43], the authors propose a novel deduplication algorithm in which a given file is broken into multiple segments. Each segment is encrypted by a user, and the encryption process involves both a secure hash function and a block encryption technique. These segments have an index tree, which is composed of hash values of such segments. The index tree is generated and encrypted by the user using an asymmetric algorithm. The authors claim that their approach will prevent storage providers, if it is implemented, from getting access to users' data or their decryption keys. In [44], the authors propose a new deduplication technique based on Attribute-Based Encryption (ABE) to encrypt data stored in the cloud and at the same time provide a secure data access to such data. According to authors' evaluation, this technique is suitable for practical deployment due to its effectiveness, scalability, and efficiency. Other research proposals associated with this topic can be found in [45,46]

For interested readers, the recently published survey regarding this topic can be found in [47].

***(MT2)* Secure storage schemes:** Attributed to attack *AT3*, *AT4*, *AT8* and *AT10*. Secure storage techniques can be used to prevent IoT data breaches. To this end, several research proposals have been introduced classified broadly into two categories: (i) cryptographic-based scheme and (ii) non-cryptographic-based scheme. An example of cryptographic-based scheme can be found In[48] .The authors have proposed a secure IoT storage technique in which IoT aggregated data can be stored securely on an object based on Shamir's secret sharing method. To protect IoT data on such system, Shamir's secret sharing algorithm has been used, along with internal padding. Data in this approach, prior to storing it, is divided into many segments and each segment will be stored in different storage objects. Another example of cryptographic-based scheme is presented in [49]. The authors proposed a new technique based on an elliptic curve algorithm, which allows different users to access and store their data securely from the cloud. Moreover, it assures that neither cloud storage provider nor unauthorised users can get access the data. This approach is also capable of protecting individuals' data, even when the cloud provider is compromised due to its data encryption.

In [50], the authors propose a novel architecture in which individuals and companies can upload and store securely their data in the cloud. Such architecture is composed of things, gateway, network infrastructure, and cloud. To collect data in such architecture, IoT objects or things are deployed in physical environment. The need of gateways in this architecture, which uses as an intermediate layer between objects and cloud, stems from that not all IoT objects are equipped with connectivity protocols(e.g., Wi-Fi) that allow them to connect to the Internet and transmit their data. The administrator in such system plays a key role in defining responsibilities according to the job functionality described in the organisation. For the interested readers, other research efforts associated with this topic can be found [51–54].

An example of non-cryptographic-based scheme can be found in [55]. The authors have proposed a new storage schema called POST-SHAREDS which offers long-term security for IoT data without involving any encryption techniques. The security of such scheme derives from dividing data into so many segments (each segment has its own pointers) and scattering them into different storage. If an attacker wants to get data of one segment, he needs to get all its pointers, which are distributed in several storage.

***(MT3)* Access control:** Attributed to attack *AT6* and *AT8*. Attributed to attack *AT3*. To control access to stored IoT data by customers or companies, many research efforts have been proposed. Such efforts can be divided broadly into four categories: (i) Mandatory Access Control (MAC), (ii) Discretionary Access Control model (DAC), (iii) Role Based Access Control model (RBAC) and (iv) ABE. Having integrated MAC into an IoT system, the system administrator will have privileges to manage the customers' roles as well as rights. In MAC, it is also possible that the system administrator manipulates access policies resulting in preventing customers to access the system. This type of access method can be added to sensitive systems like military and research centres[56]. If DAC is integrated into an IoT system, the customers will have rights to manipulate the access rules for any objects. This approach is extremely dangerous if an attacker gains access rights over a customer account. Thus, it is not wise to give one customer full rights to the IoT system.

If RBAC is integrated into an IoT system, customers can gain access to resources based on their roles and responsibilities in the system. Several research proposals have been conducted in relation to this topic [57,58,58,59]. For instance, the authors in [59] have suggested new five principles known as Abstraction, Separation, Containment, Automation and Accountability (ASCAA) for the next-generation RBAC. They claimed that such principles are applied to access control in general despite the fact that they are developing specifically to RBAC. ABE provides adaptable one-to-many encryption without prior information who will be accessing such information. It also draws attention for fine-grained access technique over outsourced data. The identification of a customer in ABE is accomplished by a set of attributes which can be used to define access policy of the customer [60]. Recently, several research proposals have been attempted to implement ABE in fog computing [61–63]

***(MT4)* Recovery Strategy:** Attributed to attack *AT3* and *AT5*. Despite the importance of providing high availability and disaster recovery for IoT storage, a few research proposals have been found in the state-of-the-art. In [10], authors have investigated the problem of uploading IoT data from a set of several sensors and the creation of different replicas of this data on distributed storage in the cloud. The applicability of such approach depends on the existence of several distributed data centres known as mini-clouds. In [64], authors propose a new replication approach to minimise power consumption, delay, and the cost of uploading a huge amount of data send by several IoT applications. Each application is composed of too many small objects. To reduce time latency, the authors deployed local cloud computing resources. Other research proposals related to this topic can be found in [65–67]

***(MT5)* Anonymisation schemes:** Attributed to attack *AT2*, *AT7*, *AT8* and *AT9*. Such solutions can fall broadly into three categories: (i) K-anonymity, (ii) L-diversity and (iii) (iii)T-closeness.

K-anonymity is a technique in which the privacy of data holders will be preserved when they issue their data, preventing threats associated with subject identification. This technique assures that the information for each person cannot be identified from a set of at least $k(-1)$ individuals who belongs to. The concept of *k*-anonymity represents data as a table composing of set of rows and columns. Each row indicates the insertion of new information related to specific entity and it should not be unique [68], whereas each column represents an attribute for entity. Two techniques have been used to achieve *k*-anonymity. First is suppression in which the values of some attributes are substituted by an asterisk $*$. Second is generalisation in which the personal values of attributes are changed by values in a wider range. For instance, if the attribute *age* is used, the value of 35 can be substituted by the term < 40.

In IoT, *k*-anonymity can be used for the localisation of smart objects to enhance location privacy. This can solve security issue related to the need of a third entity for managing different *k*-anonymity sets for several queries, inapplicability of using universal GPS regulates indoor, and obfuscation. In [69], the authors proposed a tree based location privacy technique against multi-precision attacks using a new location query technique in which multi-precision queries are fully supported. In [70], the authors propose another *k*-anonymity technique in which data can be released based on concrete generalisation.

L-diversity is suggested to mitigate *k*-anonymity weakness which is its inability to prevent homogeneity attack as well as background attack. In [71], the authors propose a new and powerful privacy technique known as *l*-diversity which can be used to prevent several attacks(e.g., homogeneity attack). Moreover, they perform an experimental evaluation to show that the proposed technique is practical, and it can be implemented effectively.

T-closeness was first coined in [72] to overcome the shortcomings of *k*-anonymity and *l*-diversity associated with attribute inspiration. The authors in [72] propose that a distribution of sensitive information in any set must be close or connected to their scattering in the whole database. To summarise the value of such work, the authors have used different real examples as well as experiments. In [73] , the authors suggest a decomposition technique with $(n-1)$ closeness, the main purpose of which is to preserve privacy in case of several sensitive attributes by reducing the amount of sensitive information which may be elicited from the published data in the *t*-closeness situation.

***(MT6)* Transient data storage:** Attributed to attack *AT1* and *AT4*. The existing research proposals have been focused on managing the persistent data in IoT systems. In this case, data may be stored even after such systems have finished their executions. Nevertheless, a handful of research works have been concentrated on managing transient IoT data generated during systems executions. The importance of transient data stems from processing data during system execution to generate new visions of data, which may be stored in storage for users' needs or may be purged, and therefore it may reduce threats associated with such data. In [74], the authors have proposed a new system of managing transient IoT data in which such data can be processed, placed and managed. This system is composed of several components like resource estimator, transient data characterisation, and data manager. Other research proposals related to this topic can be found in [75–77].

***(MT7)* Searchable Encryption (SE):** Attributed to attack *AT5* and *AT6*. Another way to protect data in IoT storage is to perform information retrieval on encrypted data which is known as a SE boomed in 2000. The main idea behind such technique can be summarised as follows: An object should index and encrypt its data, and then it sends its encrypted data along with an index to a server. In order to search for given data, the object needs to generate a trapdoor thought which the server can execute search operations directly on encrypted data, and the output will be encrypted too. This field is known as a homomorphic encryption. In this regard, Fully Homomorphic Encryption (FHE) has been proposed in 2009 by Gentry[78]. That said, the key distribution as well as user revocation in a multi-user search setting is a need and a challenge at the same time. To this end, some traditional technologies like broadcast encryption proposed in [79] and secret sharing suggested in [80] can be

used to cope with the key distribution issue. While user revocation can be solved using either trusted third party proposed in [81] or semi-trusted third party suggested in [82].

It is worth mentioning that ABE proposed in [83] has been used by Sun et al. to develop the Attribute-Based Keyword Search (ABKS) scheme to offer fine-grained search authorisation in the cloud, proposed in [84,85].

***(MT8)* Distributed data repositories:** Attributed to attack *AT5*. Several research studies have been proposed related to this topic. In [86], authors have proposed a new secure storage scheme for sharing data in public storage (in the cloud) known as a Shield. Both authentication and access control in such approach are granted by a proxy server. A new version of Merkel Hash Tree was introduced to achieve integrity check and file content update. Moreover, both key management and effective permission revocation can be accomplished using a hierarchical key organisation. Another example of integrating access control into secure storage can be found in [87]. The authors propose a new secure storage repository called Cryptonite for sharing a huge amount of scientific data in the cloud. This approach provides an easy way for its users to securely store and share their data in the cloud without revealing their sensitive data not only for unauthorised users or attackers but also for the cloud storage provider and system itself.

It is also worth mentioning that a secure version of Hadoop Distributed File System (HDFS) can be used to achieve such objective, and research proposals associated with this topic are described below:

In [88], the authors have proposed a secure version of HDFS in which two security countermeasures are involved to prevent hackers getting data in the cloud. The first countermeasure is a trust mechanism between name node used to manage data nodes and end user is established. This type of trust mechanism requires that end user must be authenticated in order to access name node. To achieve such objective, the end user first sends hash function, and then name node compares hash functions, which are Secure Hash Algorithm 2 (SHA-2), generated by both the end user and the name node. The end use is only authorised to access system if compare result is correct. The other countermeasure is random encryption methods like Rivest–Shamir–Adleman (RSA), Advanced Encryption Standard (AES) and Rivest Cipher 6 (RC6) are used on data to prevent an adversary to get access such data. The encryption and decryption process is accomplished by MapReduce which allows data aggregation and parallel processing of a huge amount of data.

Another example of secure HDFS which is equipped with three countermeasures can be found in [89].

***(MT9)* Introspection:** Attributed to attack *AT5* and *AT6*. Another technique which can be used to conserve users' sensitive information is the introspection by checking all the activities on virtual machines (VMs) in which IoT data is stored. The main idea behind such technique is to inspect the state of Central Processing Unit (CPU) for each VM, detect the malicious software on VM, and check Input & Output (IO) for records or files. Nevertheless, users' privacy may be compromised as a consequence of losing one object's integrity by malicious software.

***(MT10)* Blockchain:** Attributed to attack *AT2*, *AT6* and *AT8*. The use of blockchain technology in IoT has several advantages, the most dominant of which are decentralisation, trust, and non-repudiation. It is clear that previously mentioned countermeasures can be used to solve different security and privacy issues (e.g., unauthorised access and data leakage) associated with IoT data at rest. Yet there is a need of blockchain technology to address other important issues like untrusted TPAs and data integrity across different cloud storage. A handful of research proposals have been proposed to contribute to such objectives, the recently published of which can be found in [7,90–92]. In [7], the authors have proposed a blockchain-based solution to provide decentralised process in which data integrity for IoT data stored in the semi-trusted clouds is verified and checked. Furthermore, the

authors have illustrated the feasibility of their approach by applying a proof of concept on a personal (private) blockchain system. In[90], the authors first proposed three different requirements to allow IoT systems to share and store their data in such an untrustworthy environment like untrusted TPAs. Such requirements are divided into three categories: (i) trusted trading, (ii) trusted privacy and (iii) trusted data access. Moreover, they proposed a decentralised architecture based on blockchain technology to accomplish above mentioned requirements. The authors also demonstrated the feasibility of such technique by implementing a proof of concept on Ethereum blockchain. In [91], the authors have suggested a data-centric approach based on blockchain technology which concentrates on sharing, resilience, and auditable preservation of data. However, the authors have only presented the initial design of their approach, which is blockchain-based end-to-end encrypted data storage system. A secure and permanent data management is achieved as a result of using blockchain as an auditable access control level to a distributed storage level. In [92], the author have proposed a blockchain-based storage system called Sapphire. This system is designed specifically for data analytic in IoT. In such system IoT data coming from different IoT applications like smart home, smart grid,and smart city is classified into two categories, namely text data and media data. This kind of classification is accomplished by a data classifier. The collected data in both formats either text or media is stored into blockchain-based a large scale of storage via customer process. Each IoT object, in Sapphire, is represented as an Object-based Storage Device (OSD). Sapphire links the system interface model through the Put/Get application program interface. The main building block of Sapphire is an EB-scale storage system which uses the hash-based mapping approach to divides the key address space into OSDs. The OSDs are used as a technique to enhance load balancing and more importantly to simplify the cooperative caching. The number of OSDs may be scaled up or down in size based on the number of physical objects which may joint or depart the system. To investigate the issue of fault tolerance caused by storage nodes failure, several data replicas are used.

***(MT11)* Physical security:** Attributed to attack *AT8*. IoT data may be scattered in different physical locations making them susceptible to physical attacks despite the the existence of previously-mentioned solutions. Therefore, there is a need of physical security measures for protecting IoT data at rest. This is because the above mentioned solutions can not prevent the physical damage of IoT objects along with their storage as well as data centres. To date, several physical security solutions can be used to protect IoT data at rest including, but not limited to, security guards, physical barriers, video surveillance, and locks. It is also wise to improve the efficiency of such physical security measures by integrating them with IoT technology due to the use of connected sensors and actuators. Intelligent monitoring, tampering alerts, perimeter protection, and facial recognition are some examples of this kind of integration.

***(MT12)* Monitoring and auditing:** Attributed to attack *AT8*. Monitoring activities in storage of IoT data in the cloud is of paramount importance to prevent data breaches [93]. To this end, several research efforts have been conducted, some of the most recent of which can be described here. In [94], the authors have proposed a centralised monitoring technique for cloud applications used to monitor server, agents , and files along with their configurations. To overcome the limitations of a centralised monitoring approach which include scalability and most importantly single point of failure, this technique provides multi-level notifications, redundancy, and automatic healing. In [95], the authors have proposed a scalable distributing monitoring solution for clouds. Such solution depends heavily on a scattered management tree which involves a set of parameters along with their protocols for data collection. Moreover, the authors have reviewed the shortcomings of current intrusion detection solutions and also investigate the use of one of the emerging fields for securing virtual machines(VMs) in the cloud known as the virtual machine level intrusion detection. In [96], the authors proposed a novel architecture in which the virtualisation technology can be integrated into the heart of cloud computing to carry out intrusion detection security utilising hypervisor performance metrics like

packets transmitted/receive, CPU utilisation, and read/write requests. The authors also illustrate and validate that malicious activities could be happened, even when the attackers lack the knowledge of the operating system which operates within the VMs. For the interested readers, other research proposals related to this topic can be found in [97,98].

***(MT13)* Decommissioning:** Attributed to attack *AT1*. The process of proper decommissioning of IoT objects along with their data in the cloud is a fundamental requirement in IoT security, and its solutions can be broadly classified into two categories: (i) object-based solutions which focus on decommissioning of IoT objects and their on board data and (ii) cloud-based solutions which concentrate on destruction IoT data in the cloud storage. Despite the importance of object-based decommission techniques for addressing some security and privacy concerns like personal data breaches, there is a lack of research works conducted in the state-of-the-art in this regard. Nevertheless, Smart card alliance in [99] has suggested two choices for decommissioning. Firstly, the objects can be reset to factory default mode. In this option, all data in such objects will be deleted except the basic security parameters. These objects can come back to life later. Secondly, a blacklist technique implemented on a server will be used to prevent blocked objects to re-join an network unless their statuses on the server have been changed.

***(MT14)* Secure data migration:** Attributed to attack *AT4* and *AT8*. Despite the importance of secure data migration solutions in preventing some threats and attacks (e.g., unauthorised access and linkage attacks), a few research works in literature have been proposed. Next, we briefly discuss them.

In [100], the authors have proposed a secure data migration solution for migrating or transporting IoT data from a cloud storage service to another. To assure pre-migration authentication, such solution is equipped with mutual authentication composed of key splitting and sharing approaches. Having used a symmetric algorithm(RSA) to encrypt migrated data, several security goals like confidentiality, integrity and authenticity are fulfilled. Two OpenStack servers have been used to implement and validate the feasibility of such technique. In [101], the authors have proposed a simple and effective solution for migrating securely data between different cloud storage services. This technique depends heavily on the use of cryptography and steganography, and it is known as Secure Cloud Migration Architecture using Cryptography and Steganography (SCMACS). To encrypt and decrypt migrated data in such technique, a shared key generated by a symmetric algorithm is used by both sender and receiver. The advantage of this approach stems from generating a dynamic value for the private key. The authors also develop a prototype to illustrate the feasibility of their technique based on HDFS. Other techniques for migrating data securely among different cloud storage can be found in[102–104].

An overview of countermeasures proposed IoT data at rest is presented in Table 3

Table 3. A summary of the mitigation techniques proposed for IoT data at rest

Implementation Techniques	Research proposal	Year	Mechanism used
Secure storage schemes	[48]	2015	Suggests a secure and scalable IoT storage technique that meets different non-functional requirements (e.g. flexibility, liability, and security)
	[50]	2016	Proposes a technique in which IoT data can be stored securely using cryptographic algorithms and several access control policies
	[51]	2018	Proposes a flexible framework to address storing IoT data securely by merging cloud computing used to store non-time-sensitive data and fog computing used to store time-sensitive data.
	[53]	2016	Provides possible techniques to address some cloud computing issues like data breaches, unavailability, and reliability

	[52]	2014	Proposes a file system based on Ciphertext-Policy Attribute-Based Encryption (CP-ABE) in which secure deletion as well as access to encrypted files in the cloud can be achieved
Access control	[60]	2017	Suggests a hybrid method composed of data encryption with fine-grained access technique and index encryption, which is suitable for a fog computing ecosystem
	[61]	2016	Proposes security model called Chosen Ciphertext Attack (CCA) of attribute-based encryption with outsourced decryption, which is the best technique for IoT data protection in the cloud
	[63]	2018	Proposes a general framework in which a fully secure leakage-resilient function encryption technique is constructed to prevent several attacks (e.g., side-channel attacks)
	[57]	2012	Discusses different solutions like encryption, authentication interface, and multi-level virtualisation, which can be used to protect user data
	[62]	2018	Suggests a novel technique to reinforce CP-ABE solutions to offer immunity against key-delegation abuse concern.
Distributed data storage	[86]	2014	Proposes a new type of the Merkle Hash Tree to enhance effective integrity checking, secure file sharing without any alteration, and file content update.
	[105]	2011	Suggests a novel approach for a secure data repository service developed on a public cloud infrastructure to allow customers to securely share and store their sensitive data in the cloud
	[88]	2015	Describes HDFS and its difficulty in preserving the security and privacy of Big data
	[89]	2015	Represents three techniques (Kerberos, Name node, and Algorithm) used to enhance HDFS security
Recovery Strategy	[10]	2017	Suggests distributed cloud storage system based on mini-clouds assumption used to replicate and upload IoT data among different data centres
	[64]	2016	Suggests the distribution of local cloud computing resources to investigate the long-term evolution architecture shortcomings within radio access network and proposes a new protocol for a memory replication
	[65]	2013	Proposes a divide-and-conquer technique which uses to replicate the content in Wireless Mesh Networks (WMN)
	[67]	2015	Suggests different local search algorithms and represents a new technique called Aurora used to apply such algorithms in HDFS with lower overhead
	[66]	2013	Suggests a lightweight and scalable approach in which object replication and placement technique in WMN is achieved
Transient data storage	[74]	2018	Proposes a technique for managing mass transient data in IoT applications, which consists of several components (e.g., transient data characterisation, data manager, and resource estimator)
	[75]	2014	Proposes a software architecture that facilitates the collection of sensors' data in IoT environment
	[77]	2015	Proposes a distributed cloud-based storage system developed specifically for IoT data
	[76]	2014	Propose a new approach known as IoT4S used to collect environmental data from different heterogeneous objects to be stored in a data storage system that supports SQL and NoSQL technologies
	[69]	2012	Suggests an anonymity-tree based technique suitable for IoT, which reinforces multi-precision queries

	[70]	2012	Proposes an enhanced fine-grained algorithm FK-anonymity which re-evaluates the generalisation scale corresponding to application domain
Anonymisation based solutions	[72]	2008	Proposes a new privacy notation known as t-closeness in which the distribution of a feature in any class should be very close to the distribution of the feature in the whole table
	[73]	2010	Proposes a privacy procedure based on information theory, the implementation of which depends on the postrandomization method.
	[79]	2011	Suggests some improvements in searchable symmetric encryption in terms of effective constructions and definitions
	[80]	2008	Addresses the notation threshold privacy preserving keyword search (TPPKS), identifies its security goals, and develop a TPPKS approach.
Searchable Encryption	[82]	2011	Proposes a multi-user searchable encryption approach which is more practical, and it has a set of benefits over the popular techniques
	[84]	2016	Proposes the first attribute-based keyword search approach equipped with an effective user revocation technique, offering fine-grained search authorization
	[90]	2018	Proposes a decentralised architecture based on block-chain technology in which the trust and integrity of IoT data among TPAs can be achieved
	[7]	2017	Proposes a blockchain-based architecture to provide the integrity of IoT data among different TPAs
Blockchain based solutions	[91]	2017	Proposes a block-chain IoT data storage system which provides a secure and permanent IoT data management
	[92]	2018	Proposes a new IoT data storage system (known as Sapphire) based on block-chain technology. IoT data coming from different IoT devices constitutes objects with methods, IDs, features, and policies
	[97]	2014	Proposes an approach to deal with a huge amount of data to investigate for security monitoring point of views
	[98]	2015	Proposes a new public auditing technique based on Merkle Hash Tree known as MuR-DPA which integrates a new authenticated data structure(ADS)
Monitoring and auditing	[96]	2014	Suggests a hypervisor-based cloud intrusion detection technique that does not need extra software installed in VMs and , at the same time, provides more benefits compared to host-based intrusion detection techniques
	[95]	2013	Proposes a scalable monitoring approach for clouds to supervise their data in a distributed manner
	[34]	2010	Suggests simple solutions which allow cross-user deduplication and, at the same time, minimize the danger of data leakage.
	[43]	2012	Proposes a secure data deduplication architecture for cloud storage
Deduplication schemes	[45]	2013	Proposes a secure and effective storage service known as ClouDedup, which enables block-level deduplication as well as data confidentiality
	[44]	2016	Proposes an approach based on attribute-based encryption used not only to deduplicate encrypted data stored in the cloud storage, but also to provide a secure data access control

	[100]	2017	Proposes a secure data migration approach to move the data from one cloud storage service to another
	[102]	2015	Proposes a new architecture which enables a secure data transportation from users to the cloud server providers
Secure data migration techniques	[104]	2018	Proposes a new framework for multi-tenant cloud migration used to fulfill data integrity and confidentility
	[103]	2016	Proposes an inter-cloud data migration technique which provides better security goals and quicker response time for transferring large files into cloud storage service

5. Analysis on security and privacy guidelines for IoT data at rest

This section first describes our derived guidelines in link with the involved stakeholders. Consecutively, the overall guideline framework is presented with the linking between guidelines, mitigation techniques and attacks.

***(G1)* Minimise data storage :** The GDPR has proposed six principles for processing of personal data, among which is data minimisation. CSSs, under GDPR, should only store personal data required to achieve their processing purposes [106]. Two benefits are associated with this type of principle. One is data breach will be minimised, as unauthorised users will have access to a restricted amount of data. The other benefit is that data accuracy will be improved [107]. This guideline, hence, suggests that the amount of data stored either on objects or in the cloud should be minimised, and any segment of data that is not needed to execute a specific task should be removed from IoT storage [108]. For example, the authors in [14] have stated that raw data can be removed from storage once secondary contexts are extracted and, more importantly, all data must be de-identified. Three countermeasures, namely *MT1*, *MT5*, and *MT8*, can be used to accomplish this guideline.

MAN	DEV	PRV	CNS
✓	✓	✓	✓

Reasoning: This guideline is proposed based on one of the Privacy by design as well as Security by design principles, which is minimisation data, proposed by Hoepman in [109] and OWASP [110], respectively. We do believe this guideline can be used by manufacturers, developers, and providers as they are directly involved in the production, deployment, and development of IoT objects. This can be happened by allowing such objects to minimize the amount of data on them by deleting any segment of data that is not needed. It is also applicable to be used by customers, since in the future IoT objects may be armed with dashboard settings, permitting to minimise data collection.

***(G2)* Minimise data retention:** In [111], the authors have stated that retaining data for a long time is associated with data breaches, since it gives an attacker an opportunity to try all his/her hacking techniques to compromise it. A part from data breaches, privacy risks may also be increased, according to [14]. This is because long retention periods may cause unauthorised secondary usage. This is why the GDPR has stated that sensitive data must be stored "no longer than is necessary for the purposes for which the personal data are processed"[106]. This guideline, therefore, suggests that data retention on IoT objects or in the cloud should be minimised as possible. This guideline can be implemented by *MT6*.

MAN	DEV	PRV	CNS
✓	✓	✓	✓

Reasoning: This guideline proposes based on a minimisation principle suggested Privacy by design (Hoepman in [109]) as well as Security by design (OWASP in [110]) frameworks. It can be utilised by manufacturers to ensure that their objects are equipped with data retention rules. For providers and developers, this guideline can be implemented to ensure that IoT applications are engineered from start to avoid keeping data longer than it is required. Customers can also benefit from this guideline by deleting unnecessary data on their objects.

***(G3)* Distributed data storage:** To prevent a single point of failure in IoT applications and enhance their availability, this guideline suggests that IoT data should be stored in a distributed manner. However, using such guideline has trade-offs. On one hand, it improves the availability of IoT applications and also reduces some privacy risks. For instance, the authors in [14] have stated that the use of distributed data storage can be used to minimise privacy violations, since it prevents unauthorised access and secondary knowledge discovery. On the other hand, it opens doors for several attacks and threats like data leakage, as it increases the attack surface of IoT, according to [112]. Therefore, this guideline should be investigated with caution, and it can be implemented by *MT8*.

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning: This guideline is suggested based on aggregation principle proposed in privacy by design framework (Hoepman in [109]). It can be utilised by manufacturers to assure that their IoT products have capabilities to store data in distributed environments. For providers and developers, it can be utilised to assure that their IoT applications and services are designed to store data in distributed environments by supporting distributed IoT architectures.

***(G4)* Encrypt data storage:** In [17], the ENISA has expressed the importance of encrypted IoT data at rest to minimise privacy violations in IoT. It is also worth mentioning that the Payment Card Industry Data Security Standard (PCI DSS) has forced all companies dealing with credit card information like Visa and Master Card to implement encryption techniques when storing data [113]. Moreover, PCI DSS explicitly prevents the use of storage encryption as provided by operating systems. This guideline, thus, suggests that data of IoT applications should be stored in encrypted manner either on IoT objects or in the cloud. Two protection measures (*MT2* and *MT7*) can be used to achieve such guideline.

MAN	DEV	PRV	CNS
✓	✓	✓	✓

Reasoning: This guideline is stated based on hide principle proposed in privacy by design framework (Hoepman). It can be utilised by manufacturers to assure that their IoT products have capabilities to encrypt their stored data. Both provider and developers can integrate such guideline from start into their IoT applications so that they always store their data in encrypted format. Customers can also benefit from this guideline by enabling this feature if it comes with IoT products.

***(G5)* Prevent data leakage:** Even though if IoT data is stored in an encryption form, it is still vulnerable to side-channel attacks. This can be happen due to many reasons such as weak encryption, hardware failure, and human error[53]. Therefore, this guideline suggests that IoT stakeholders like manufacturers and providers should implement suitable data leakage prevention techniques (e.g., SE techniques) by taking into their considerations governments rules and industry standards. Six implementation techniques(*MT1*, *MT2*, *MT5*, *MT6*, *MT6*, and *MT12*) can be used to carry out this guideline.

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning: This guideline is suggested based on hide principle proposed in privacy by design framework (Hoepman). Manufacturers can integrate such guidelines into their IoT products to be more resistant to side channel attacks. It can also be utilised by providers and developers to assure that their IoT applications are engineered from ground up to prevent data leakage.

***(G6)* Minimise data granularity:** The granularity here refers to the level of details available and to be utilised. A concrete representation of data (e.g., a full address) is known as high granularity, while a summary view of data or a high level representation of data is called low granularity (e.g., a dissemination of location), according to [114]. In this context, storing concrete data is always associated with high privacy risks compared to storing data in a abstract level, since it is composed of more data. Therefore, this guideline suggests that IoT systems should only store minimal level of data in which their functions can be maintained [14]. This guideline can be implemented by *MT1* and *MT7* .

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning: This guideline is stated based on a minimisation principle suggested in Privacy by design (Hoepman in [109]) as well as Security by design (OWASP in [110]) frameworks. It can be implemented by manufacturers to ensure that their products always collect and store information about their customers in a high level. Both providers and developers can also integrate such guideline into their IoT applications from ground up to prevent identification attacks as they store data in a abstract level .

***(G7)* Ensure data availability:** With the growth of IoT, data will be generated at an unprecedented rate, as billions of objects will be connected to the Internet. Since most of these objects like actuators, sensors, and thermostats do not have on-board storage, their data must be stored in cloud data centres [10].The availability of such data is a crucial requirement for so many applications to achieve their tasks. This guideline, therefore, suggests that CSSs as well as IoT objects should implement efficient techniques (e.g., recovery strategy and DoS prevention) by which the availability of their data is guaranteed in case of natural disasters or some attacks. For instance, an attacker could flood a storage server with invalid data at very high rate in such a way that the storage server wastes most of its time validating the authenticity of data and, therefore, fails to timely respond to valid network traffic [31]. This guideline can be implemented by two protection measures (*MT4* and *MT1*).

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning:This guideline can be utilised by developers, providers, and manufacture to ensure that their applications, services, and objects are always accessible by their authorized users.

***(G8)* Location-based aggregation:** This guideline suggests IoT applications should aggregate their data based on geographical boundaries. For instance, a query would be ”how many medical objects used in each city in Switzerland”. The response to this question would be an aggregated value which is unique to each city". However, it is not needed to gather details about individuals medical objects, as it may lead to privacy breaches [115]. This guideline can be implemented by *MT8*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is suggested based on aggregation principle proposed in privacy by design framework (Hoepman in [109]). Manufacturers and providers can implement this guideline into their products and services so that they can aggregate their data based on geographical boundaries.

***(G9)* Post-inform customers:** A recently published survey by Eurobarometer has shown that 67 percent of Europeans indicated that they were worried about their information and personal data that they offer online, since they lack control over it [22]. To this end, GDPR was developed to give the individuals of EU a control over their personal data. This guideline, thus, suggests that IoT applications should always inform their users before storing or sharing data related to them. For instance, GDPR states that processing the personal data of individuals in business enterprises requires an explicit opt-in or consent from them, and several kinds of data will necessitate distinct consent [106]. This guideline can be implemented by *MT2*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on inform principle proposed in privacy by design framework (Hoepman in [109]). This guideline can be implemented by manufacturers to assure that their objects have capabilities to infrom their customers about their data. Providers also can implement this guideline into their services so that they can notify users, when they store personal data or sensitive data, and give users control over their data

***(G10)* Chain data aggregation:** This guideline suggests to accomplish data aggregation on-the-go as data transfers from one object to another[14] so that each object will have an opportunity to respond. In this case, if there is a query (e.g., requires a count), any object can respond to it without the need of a centralised entity. This will give all objects chance to respond to such query, and, at the same time, it will improve availability of IoT [116]. This guideline can be implemented by *MT8*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is proposed based on aggregation principle proposed in privacy by design framework (Hoepman in [109]). Both manufacturers and providers can implement this guideline into their products and services so that they can aggregate their data while transferring data from one object to another.

***(G11)* Time-Period data aggregation:** This guideline suggests that IoT applications should store their data over a long period of time like days and months. This guideline will minimise the granularity of IoT data which, in turn, reduces data breaches. For instance, it is wise to report power consumption of a given building in aggregate form per week rather than a daily basis [117]. This guideline can be implemented by *MT8*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on aggregation principle proposed in privacy by design framework (Hoepman in [109]). It can be implemented by manufacturers to assure that their products are armed with techniques in which customers can decide when they want such product to aggregate data related to them. Providers also can integrate this guideline into their services so that they can aggregate over a long period of time.

***(G12)* Ensure authorised access to IoT data:** The importance of providing solid techniques to control access to IoT data at rest derives from two primary factors. One is an IoT object may require to communicate with other objects in order to share and access their data. Such object, therefore, must only interact with authorised objects. The other factor is that different IoT objects may store their data in the CSSs which are only logically isolated, but in reality such data may be physically kept in the same data centre [118]. Overlooking access control mechanisms to IoT data at rest may lead to harmful consequences. For instance, the authors in [6] show that a thermostat deployed in a smart home depends heavily on a smoke detector's data to turn a heating system off in case of danger. However, sharing and accessing the smoke detector's data by unauthorised objects may place the whole smart home at risk. This guideline, thus, suggests that each IoT object or CSS should be armed with authorisation techniques (e.g., a role-based technique) through which all unauthorised requests are blocked or prevented. Three implementation techniques, namely *MT11*, *MT9*, and *MT3*, can be used to fulfil this guideline.

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning: This guideline is suggested based on principle of defense in depth proposed in security by design framework (OWASP [110]). This guideline can be implemented by manufacturers to ensure that their products are equipped with measures via which only authorised users can get access to them. Providers as well as developers also can integrate this guideline into their services and applications so that only legitimate users can gain access to their stored data.

***(G13)* Remove or hide sensitive data:** This guideline suggests that IoT applications must first get rid of a personally identifiable information and, then, store it. It is obvious that storing data set along with its personally identifiable information will significantly increase the risk of privacy losses. For instance, the authors in [119] have illustrated that a retailer company, called Target, once received a complaint from a customer who was really disappointed, as company sent coupons for kids' clothes to his teenage daughter. Nevertheless, the Target intentionally sent such coupons to the daughter since she was pregnant at that time. This type of inference may happen as a consequence of storing data along with its personally identifiable information, which, in turn, helps such company to conduct data mining on its customers' data. This guideline can be implemented by *MT5*.

MAN	DEV	PRV	CNS
✓	✓	✓	✗

Reasoning: This guideline is proposed based on hide principle proposed in privacy by design framework (Hoepman in [109]). Manufacturers can integrate such guideline into their products to ensure that they have capabilities to de-identify personal data before storing it. It can also be implemented by both provider and developers to assure their applications and services are developed from ground up so that they are capable of identifying a personally identifiable information and more importantly de-identifying it before storing it.

***(G14)* Search on encrypted data:** As several research efforts have expressed the importance of performing information retrieval on encrypted data to prevent data linkage attacks [17,18,21], this guideline suggests that IoT applications should be shielded with techniques (e.g., SE) that allow IoT applications to respond to any queries by searching on encrypted data without revealing sensitive information. This guideline can be implemented by *MT7*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on two principles, the first of which is hide principle proposed in privacy by design framework (Hoepman in [109]). The second principle is defense in depth suggested in Security by design framework (OWASP in [110]). Manufacturers and Providers can implement such guideline into their objects and services to assure that such objects and services have capabilities to search on encrypted data to alleviate linkage attacks.

***(G15)* Provide Data Integrity across different platforms:** With the emergence of IoT, different IoT applications, which may store their data on several platforms, may cooperate with each other to accomplish specific tasks or services. If data integrity of a single IoT application in the cloud has been tampered with, there is a risk to deal with unsecured applications [7]. This guideline, therefore, suggests that IoT objects as well as CSSs should be equipped with techniques in which data integrity across different platforms must be checked. This guideline can be implemented by *MT10*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on not trust principle suggested in Security by design framework (OWASP in [110]). This guideline can be implemented by manufacturers and providers to ensure that their objects and services are capable of checking the integrity of data they deal with.

***(G16)* Secure share data with untrusted TPA:** Secure share data among untrusted TPA is of paramount importance for two reasons. One is IoT systems will store their data on different CSSs due to their limited capabilities as well as dynamic nature of such technology. The other reason is that it is unrealistic to assume that all cloud service providers are reliable as expected. Thus, this guideline suggests that IoT systems should be armed with techniques in which such systems could store and share their data securely in different CSSs , even when some of which are not untrusted[92]. The guideline can be implemented by *MT10*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on two principles, namely not trust and defense in depth suggested in Security by design framework (OWASP in [110]). Both manufacturers and providers can equip their objects and services with techniques through which such objects and services can share their data securely among untrsted TPA.

***(G17)*Prevent physical access to data storage:** In [120], the authors stated that physical attacks against traditional computers have become more easily to carry out compared to logical attacks, since logical security measures have been significantly improved. Likewise, IoT will suffer from physical attacks. This is because IoT inherits most of the issues of the existing Internet and, most probably,

increases them due to direct association with physical objects [4,21]. Hence, physical security in IoT is crucial due to the fact that logical security measures like firewall, intrusion detection systems, and encryption can not prevent physical attacks against IoT objects as well as data centres in the cloud. This guideline, thus, suggests that a barrier around IoT objects and data centres should be placed to prevent unauthorised physical access. This guideline can be implemented by *MT11*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is proposed based on control principle suggested in Privacy by design framework (Hoepman in [109]). This guideline can be utilised by manufacturers to equip their products with techniques to prevent physical tampering. Providers can also benefit from such guideline by storing data collected by their provided services in different data storage which may be located in environments, over which they have control.

***(G18)* Take precautions in case of natural disaster:** In [121], the authors have stressed the importance of taking precautions in case of natural catastrophe. To this end, they have used three backup servers, the data of which must be stored in encrypted format. If something unusual occurs to the server, the secret key used to encrypt data will be used again to decrypt it. This guideline suggests that CSSs should have recovery strategies used to get their data back in case of unusual situations. This guideline can be implemented by *MT4*.

MAN	DEV	PRV	CNS
✓	✓	✓	✓

Reasoning: This guideline is proposed based on fail securely principle proposed in Security by design framework (OWASP in [110]). This guideline can be utilized by all stakeholders by implementing recovery techniques so that they have a copy of their data in case of natural disasters.

***(G19)* Minimise duplicated copies:** Unlike minimising data storage which focuses on removing unnecessary segments of data not required to carry out a specific task before storing phase, this guideline concentrates on minimising duplicate data in the cloud. This kind of data replication could occupy network bandwidth and may be stored in different data storage, increasing the attack surface. Data duplication in the cloud derives from two main factors. One is HDFS generates a lot of duplicate data due to its a replication mechanism [31]. The other factor is different IoT objects may be deployed to monitor the same environment which may generate duplicated copies of IoT data [122]. This guideline, therefore, suggests that cloud-based storage services should be equipped with a technique in which only a unique copy of duplicate data is stored [47]. This guideline can be implemented by *MT1*.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on minimising attack surface area principle suggested in Security by design framework (OWASP in [110]). it can be utilised by both manufacturers and providers to guarantee that their products and services only have a distinct copy of duplicate data, and most importantly they store in different locations from its origin.

***(G20)* Proper data destruction:** The secure destruction of IoT data either in IoT objects or in the cloud is a vital requirement to prevent different security and privacy issues (e.g., data leakage). In

one hand, the destruction of IoT objects along with their sensitive data is inevitable, since each IoT object will reach its end of life, and therefore its data must be destroyed properly [99]. In this case, this guideline suggests that each object should be equipped with a clear end-of-life technique in which such object can be disposed or destroyed without exposing its sensitive data [123]. On the other hand, the destruction of IoT data in the cloud stems from many reasons. One is due to the termination of a contract with a provider in which a secure deletion of customers' data must be accomplished [124]. Another reason is because of the compliance to GDPR in which people have the right not only to to access their personal data, but also to demand the destruction of their data [106]. In this case, this guideline suggests that data providers should delete and stop further use of users' personal data as they ask for their data to be forgotten.

MAN	DEV	PRV	CNS
✓	✓	✓	✓

Reasoning: This guideline is suggested based on control principle suggested in Security by design framework (OWASP in [110]). It can be utilised by manufactures to ensure that their objects have capabilities to destroy their data in a secure manner when they reach an end-of-life stage. Providers and developers can also integrate this guideline to their services and applications in order to give their users control over their data, among which the right of their data to be forgotten.

***(G21)* Secure data migration:** Although most IoT systems transfer users' data to the cloud storage services due to their limited capabilities in terms of memory and storage space, such systems may decide to migrate or transfer their data from one cloud storage to another due to many factors (e.g., the lack of security and availability)[125]. This process of migrating or transporting data is known as data migration, and it must be carried out in a secure manner. That said, the lack of secure data migration when moving data from one CSS to another may open a door for many attacks and threats. For example, the authors in [126] have stated that if data migration is not accomplished in a systematic and proper manner, such data is susceptible to many attacks like unauthorised access. Hence, this guideline suggests that CSSs should be armed with techniques in which data migration among them should be carried out in a secure way.

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on not trust services principle proposed in Security by design framework (OWASP in [110]). It can be utilised by providers when they transfer customers' data from one data storage to another. Manufacturers also can benefit from such guideline when moving their data (e.g., objects update) from one serve to another.

***(by G22)* Manage encryption keys:** To protect sensitive data, strong encryption is of paramount importance. Although several research proposals have been proposed to encrypt IoT data, the lack of strong technique for managing the encryption keys is a common issue among them, which may lead to several vulnerabilities like unauthorised access. Moreover, the process of managing thousands of encryption keys within an IoT company is a challenge. This guideline, therefore, suggests that encryption keys must be kept on separate devices from the data they are used to encrypt. This kind of separation makes it harder for attackers to compromise data and its encryption keys at the same time [127].

MAN	DEV	PRV	CNS
✓	✗	✓	✗

Reasoning: This guideline is stated based on defense in depth principle proposed in Security by design framework (OWASP in [110]). This guideline can be implemented by both providers and manufacturers to ensure the protection of users' data, since they separate encryption keys from encrypted data.

Fig. 1 summarises the relationship between our proposed guidelines for IoT data at rest, followed by their suitable mitigation techniques, and associated attacks vectors.

6. Discussion and future work

A summary of the previously mentioned research efforts is presented in Table 4, along with our intended objectives. It is not hard to observe many limitations while going through them. Our research, therefore, is directed to overcome those shortcomings that can be categorised as follows: (i) The absence of a comprehensive list of security and privacy guidelines for IoT data at rest, followed by to whom these guidelines are intended for. (ii) The lack of suitable implementation techniques to implement such guidelines. (iii) The necessity of attack investigations related to IoT data at rest.

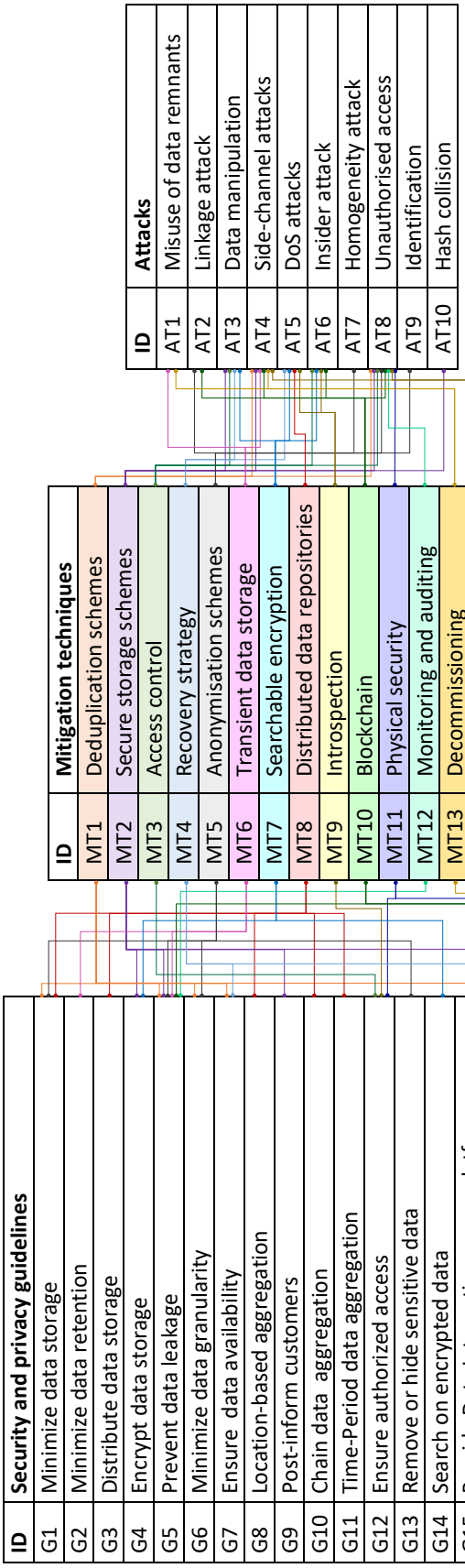

Figure 1. A summary of guidelines, stakeholders, attacks and countermeasures for IoT data at rest

Table 4. Comparison of research efforts presented in the literature.

Addressed Features		State-of-the Art Work							
		[14]	[15]	[16]	[17]	[18]	[19]	[21]	This Work
IoT security and privacy Guidelines	Minimise data storage	✓	✓	✓	✓	✓	✓	✗	✓
	Minimise data retention	✓	✗	✓	✓	✓	✗	✗	✓
	Encrypt data communication	✓	✓	✓	✓	✓	✓	✓	✗
	Secure boot process	✗	✗	✓	✓	✓	✓	✓	✗
	Hide data routing	✗	✗	✗	✓	✓	✓	✓	✗
	reduce interference	✗	✓	✓	✓	✓	✓	✓	✗
	Distribute data storage	✓	✗	✗	✗	✗	✓	✗	✓
	Encrypt data storage	✓	✓	✓	✓	✓	✓	✗	✓
	Minimise data granularity	✓	✗	✗	✗	✗	✗	✗	✓
	Location-based aggregation	✓	✗	✗	✗	✗	✗	✗	✓
	Inform customers	✓	✓	✗	✓	✓	✓	✗	✓
	Chain data aggregation	✓	✗	✗	✗	✗	✗	✗	✓
	Time-Period data aggregation	✓	✗	✗	✗	✗	✗	✗	✓
	Ensure authorised access	✗	✓	✓	✓	✓	✓	✓	✓
	Remove or hide sensitive data	✓	✓	✓	✓	✓	✓	✓	✓
	Search on encrypted data	✓	✗	✗	✓	✗	✓	✓	✓
	Provide Data Integrity across CSPs	✗	✗	✗	✗	✗	✗	✗	✓
	Secure share data with untrusted CSPs	✗	✗	✗	✗	✗	✗	✗	✓
	Minimise duplicated copies	✗	✗	✗	✗	✗	✗	✗	✓
	Prevent physical access	✗	✗	✓	✓	✗	✓	✗	✓
	Ensure data availability	✗	✗	✓	✓	✓	✓	✗	✓
	Proper data destruction	✗	✗	✗	✓	✗	✓	✗	✓
	secure data migration between CSS	✗	✗	✗	✗	✗	✗	✗	✓
	void storing enc. keys with enc. data	✗	✗	✗	✗	✗	✗	✗	✓
Types of Guidelines	Privacy	✓	✓	✓	✓	✓	✓	✓	✓
	Security	✗	✓	✓	✓	✓	✓	✓	✓
Guidelines Intended for	Manufacturer	✗	✗	✓	✗	✓	✗	✓	✓
	Developer	✓	✓	✓	✓	✓	✓	✓	✓
	Customer	✗	✗	✓	✗	✗	✗	✓	✓
	Provider	✗	✗	✗	✗	✓	✗	✓	✓
Threats Mitigated by Guidelines		✗	✗	✗	✗	✗	✗	✓	✓
Technique to implement Guidelines		✗	✗	✗	✗	✗	✗	✓	✓

6.1. Recommendations for future work

In spite of considerable research efforts devoted to the IoT security domain, we can still suggest many issues that require to be addressed.

6.1.1. Protection against insider threats

Although some of previously-mentioned protection measures like monitoring and incident response can be used to mitigate malicious insiders threats, such techniques can not fully protect IoT data at rest from such threats as they are not designed specifically for such purpose. Therefore, there

is a need to suggest a mechanism which can defend against insider threats in many IoT domains (e.g., e-healthcare organisations). This can be achieved by implementing preventive, detective, and reactive measures, such as user behavior analysis, policy-based frameworks, detection of anomalies in data access and context-based access. Such techniques must be taken into account to avoid the impacts of malicious insiders in IoT data at rest. To this end, user behavior analysis techniques have been proposed by a few number of researchers [128]. There are, however, many shortcomings in all the suggested approaches and therefore can not be implemented in all IoT domains like multi-cloud e-healthcare environment to prevent malicious insiders, according to [24]. Hence, it is a challenge for researchers to design approaches which are suitable to investigate user behavior and implement them to alleviate this threat particularly in the e-healthcare environment. It is also possible that individuals who are in contact patients could accidentally or intentionally alter the settings of IoT objects collecting health data, which may adversely impact the reply of heathcare organisation relating to patients care. It is, therefore, essential to develop an approach which could detect such a modification and at the same time notify health organisations about it so that they could turn back the changes in the settings.

Detection of anomalies in data access has been investigated in [129]. The authors propose and design a noval technique which is capable of detecting, notifying, and responding to any anomalies inside Relational Database Management Systems (RDBMS). It is also capable of automatically creating and maintaining profiles of normal applications as well as end users. This process of keeping track of normal users and applications depend heavily on their communication with monitored RDBMS during training stage. Then, it uses such profiles to identify malicious behaviour that sidetracks from normality. Nevertheless, several privacy issues may stem from monitoring the behavior of end users. Therefore, a lot of research is required to balance between security and privacy threats and end users' privacy.

6.1.2. Need for Legislation

In the literature, a huge number of issues related to IoT data are described poorly, and they need more investigation. For instance, customers would often like to know what type of data is collected and stored by their smart objects before purchasing them, which is not possible at this time. Moreover, customers may also want to know how their data stored either in the objects or in the cloud is protected. This kind of information is not generally offered. Therefore, it is also wise to give customers chance to reconfigure their privacy choices or preferences in IoT objects, in similar way as Smartphones. For instance, Google, in the mobile context(e.g., Android), has designed a dashboard technique that gives users more control over their personal data. The appropriate implementation of such dashboard techniques will provide customers more precise ways to regulate what data they want to share and when and how their personal information is gathered and used [130]. Nevertheless, further investigation is required, since the the current approaches are not mature enough for standardization, nor do they design specifically for IoT.

6.1.3. The necessity of common intercloud architecture

Despite the benefit of common interclould architecture(e.g., OpenStack that offers APIs and a framework for cloud systems) in which different clouds can coordinate, share and manage their functionalities to provide services, it still lacks a common standards which impede its interoperability. To this end, a handful research proposals have been conducted in this regard [131–133]. In [132], the authors propose a model that integrate different services to ease intercloud interaction between different platforms so as to display all available services. However,a lot of effort is required to create a graphical user interface to offer a common management platform.

6.1.4. The distribution of the cloud infrastructure over the edge computing

A new emerging technology known as edge computing has been proposed by Cisco [134] as a intermediate layer between IoT objects and cloud computing. The main objective behind such

technology is to scatter the cloud infrastructure over the edge layer, making it so closer to IoT objects and users. Several advantages in terms of bandwidth and latency that improve service quality come as a result of this closeness. It is expected that IoT may take benefit from edge cloud computing so as to achieve some desired requirements like performance and security. To this point, several research proposals are required to develop efficient security techniques based on edge computing technology. A part from required security solutions, a few questions are still raised up and need to be addressed, for instance, how to develop a trust model between IoT objects and fog nodes in such highly-scattered IoT environment.

6.2. Limitations of the study and threats-to-validity

In this paper, we propose a framework of security and privacy guidelines for IoT data at rest followed by to whom these guidelines are intended for, and their appropriate countermeasures. However, such guidelines and their protection measures are not absolute, nor they can be guaranteed the protection of IoT data at rest for three reasons. One is that new vulnerabilities are continuously being disclosed, which indicates there is a necessity to monitor, review, and maintain IoT security and privacy guidelines for IoT data at rest as well as best practices developed for particular environments and use cases (e.g., healthcare) on a regular basis. Another reason is IoT paradigm is enabled by several technologies(e.g., middleware, , sensors, and communication protocols such as a secure video steaming in [135]), and, for sure, new emerging technologies related to how either to store or process IoT data at rest will be introduced. New security and privacy guidelines, therefore, are of paramount importance to avoid IoT data breaches. Unfortunately, such rules are breakable due to the advancements of hacking tools as well as level of knowledge used by the adversaries. The other reason is that the successful of our suggested framework of security and privacy guidelines for IoT data at rest depends heavily on their implementation. Thus, the poor implementation of such countermeasures may lead to data breaches.

6.3. Conclusion

Because of unexpected growth of connected sensors and network infrastructure, the dawn of IoT in which several applications like smart car, smart building, and smart grid can interact with each other to make our lives simpler and more productive is approaching. IoT is a beneficial ecosystem that offers different solutions such as amazon echo; nevertheless, at the same time, risk associated with data breaches can be enormous too. Therefore, in this paper we conduct an in-depth analysis on IoT data at rest to identify its possible attacks and alleviate its associated risks. To this end, we propose a framework of security and privacy guidelines for IoT data at rest which can be used by IoT stakeholders who may utilise such guidelines to build secure IoT systems from the start, and, therefore, enhance security and privacy by design. This framework also shows the link between guidelines, mitigation techniques, and attacks. More importantly, we discuss our derived guidelines in link with involved stakeholders, and we also give "reasoning" under which each guideline is stated based on one or two principles of either security by design or privacy by design frameworks. Furthermore, we briefly discuss several limitations of our framework, an example of which is poor implementation of protection measures. Finally, we suggest some open challenges need further investigation.

In the future, we will propose a step-by-step methodology of how our suggested guidelines can be implemented by developers from the early stages of their IoT systems so that the poor implementation of such guidelines could be mitigated.

Acknowledgement

This work has received funding by the European Union's Horizon 2020 Research and Innovation Programme through GHOST project (https://www.ghost-iot.eu/) under Grant Agreement No. 740923 and AVENUE project (https://h2020-avenue.eu/) under Grant Agreement No. 769033. This paper

reflects only the authors views; the European Union is not liable for any use that may be made of the information contained therein.

Abbreviations

ABE	Attribute-Based Encryption
ABKS	Attribute-Based Keyword Search
AES	Advanced Encryption Standard
ASCAA	Abstraction, Separation, Containment, Automation and Accountability
CCA	Chosen Ciphertext Attack
CIA-triad	Confidentiality, Integrity and Availability triad
CP-ABE	Ciphertext-Policy Attribute-Based Encryption
CPU	Central Processing Unit
CSS	Cloud Storage Service
DAC	Discretionary Access Control model
DoS	Denial of Service
DPD	Data Protection Directive
ENISA	European Union Agency for Network and Information Security
EU	European Union
FHE	Fully Homomorphic Encryption
GDPR	General Data Protection Regulation
HDFS	Hadoop Distributed File System
HIPAA	Health Insurance Portability
IAS	Information, Assurance, and Security octave
ICC	Industrial Internet Consortium
IIoT	Industrial Internet of Things
IO	Input & Output
IoT	Internet of Things
IoTSF	IoT Security Foundation
MAC	Mandatory Access Control
OSD	Object-based Storage Device
OWASP	Open Web Application Security
PCI DSS	Payment Card Industry Data Security Standard
PHI	Personal Health information
RBAC	Role Based Access Control model

RC6	Rivest Cipher 6
RDBMS	Relational Database Management Systems
RSA	Rivest–Shamir–Adleman
SCMACS	Secure Cloud Migration Architecture using Cryptography and Steganography
SE	Searchable Encryption
SHA-2	Secure Hash Algorithm 2
TPA	Third-Part Auditor
VM	virtual machine

References

1. Terzi, D.S.; Terzi, R.; Sagiroglu, S. A Survey on Security and Privacy Issues in Internet-of-Things. *IEEE Internet of Things Journal* **2017**, *4*.
2. Rodic-Trmcic, B.; Labus, A.; Bogdanovic, Z.; Despotovic-Zrakic, M.; Radenkovic, B. Development of an IoT system for students' stress management. *Facta universitatis - series: Electronics and Energetics* **2018**, *31*, 329–342. doi:10.2298/fuee1803329r.
3. Jain, R. Internet 3.0: Ten Problems with Current Internet Architecture and Solutions for the Next Generation. MILCOM 2006. IEEE, 2006, pp. 1–9. doi:10.1109/MILCOM.2006.301995.
4. Akram Abdul-Ghani, H.; Konstantas, D.; Mahyoub, M. A Comprehensive IoT Attacks Survey based on a Building-blocked Reference Model. *IJACSA) International Journal of Advanced Computer Science and Applications* **2018**, *9*. doi:10.14569/IJACSA.2018.090349.
5. Saleem, J.; Hammoudeh, M.; Raza, U.; Adebisi, B.; Ande, R. IoT standardisation: : challenges, perspectives and solution. Proceedings of the 2nd International Conference on Future Networks and Distributed Systems - ICFNDS '18; ACM Press: New York, New York, USA, 2018; pp. 1–9. doi:10.1145/3231053.3231103.
6. Mohsen Nia, A.; Jha, N.K. A Comprehensive Study of Security of Internet-of-Things. *IEEE Transactions on Emerging Topics in Computing* **2016**.
7. Liu, B.; Yu, X.L.; Chen, S.; Xu, X.; Zhu, L. Blockchain Based Data Integrity Service Framework for IoT Data. 2017 IEEE International Conference on Web Services (ICWS). IEEE, 2017, pp. 468–475. doi:10.1109/ICWS.2017.54.
8. ENISA European Union Agency For Network and Information Security. Towards secure convergence of Cloud and IoT. Technical Report September, ENISA European Union Agency For Network and Information Security, 2018.
9. Cirani, S.; Ferrari, G.; Veltri, L. Enforcing Security Mechanisms in the IP-Based Internet of Things: An Algorithmic Overview. *Algorithms* **2013**.
10. Kumar, A.; Narendra, N.C.; Bellur, U. Uploading and replicating internet of things (IoT) data on distributed cloud storage. *IEEE International Conference on Cloud Computing, CLOUD* **2017**, pp. 670–677.
11. Riahi Sfar, A.; Natalizio, E.; Challal, Y.; Chtourou, Z. A roadmap for security challenges in the Internet of Things. *Digital Communications and Networks* **2018**, *4*, 118–137. doi:10.1016/j.dcan.2017.04.003.
12. Kim, D.; Choi, J.Y.; Hong, J.E. Evaluating energy efficiency of Internet of Things software architecture based on reusable software components. *International Journal of Distributed Sensor Networks* **2017**, *13*. doi:10.1177/1550147716682738.
13. Russell, B.; Lingenfelter, D.; Abhiraj, K.S.; Manfredi, A.; Anderson, G.; Mordeno, A.; Bell, M.; Mukherjee, V.; Bhat, G.; Naslund, M.; Boyce, K.; Nieto, J.; Cook, M.; Owen, T.; De Monts, R.; Rastogi, A.; Donahoe, T.; Sanchidrian, G.; Drake, C.; Sasahara, E.; Drgon, M.; Stenberg, J.; Futagi, M.; Subramaniyan, S.; Guzman, A.; K, T.T.; Henein, N.; Tatipamula, S.; Drew, J.H.; Duren, V.; Johnson, G.; Yeoh, J.; Hughes, L.; Pawluk, J.; Shankar, B.R.; Thriveni, S. Security Guidance for Early Adopters of the Internet of Things (IoT). Technical Report April, Cloud Security Alliance Publishing, 2015.
14. Perera, C.; McCormick, C.; Nuseibeh, B. Privacy-by-Design Framework for Assessing Internet of Things Applications and Platforms. *IoT'16* **2016**.

15. Broadband Internet Technical Advisory Group. Internet of things (IoT) security and privacy recommendations: a uniform agreement report. Technical Report November, Broadband Internet Technical Advisory Group, 2016.
16. OWASP. IoT Security Guidance.
17. ENISA. Baseline Security Recommendations for IoT in the context of Critical Information Infrastructures. Technical Report November, ENISA, 2017. doi:10.2824/03228.
18. Australia, I.A. Internet of Things Security Guideline.
19. IoT Security Foundation. IoT Security Compliance Framework. *IoT Security Foundation: Best Practice User* **2017**.
20. Trusted Computing Group. TPM Main Specification, 2011.
21. Abdul-Ghani, H.A.; Konstantas, D. A Comprehensive Study of Security and Privacy Guidelines, Threats, and Countermeasures: An IoT Perspective. *Journal of Sensor and Actuator Networks* **2019**, *8*, 38. doi:10.3390/jsan8020022.
22. SeeUnity. The main differences between the DPD and the GDPR and how to address those moving forward, 2017.
23. Dominic Chiarelli, JD, M.G. The Convergence of GDPR, the HIPAA Security Rule, and Part 11 on US Clinical Research. Technical report, Kinetiq, 2018.
24. Ahmed, A.; Latif, R.; Latif, S.; Abbas, H.; Khan, F.A. Malicious insiders attack in IoT based Multi-Cloud e-Healthcare environment: A Systematic Literature Review. *Multimedia Tools and Applications* **2018**, *77*, 21947–21965. doi:10.1007/s11042-017-5540-x.
25. Securitymetrics. An Introduction to HIPAA Compliance. Technical report, Securitymetrics, Orem, Utah, 2013.
26. ., I.I.C. The Industrial Internet of Things Volume G1 : Reference Architecture IIRA.
27. Zhang, M.; Raghunathan, A.; Jha, N.K. Trustworthiness of medical devices and body area networks. *Proceedings of the IEEE* **2014**, *102*, 1174–1188.
28. Li, C.; Raghunathan, A.; Jha, N. Hijacking an insulin pump: Security attacks and defenses for a diabetes therapy system.? in BT - Proc. IEEE Int. Conf. e-Health Netw. Appl. Serv.,. *2011 IEEE 13th International Conference on e-Health Networking, Applications and Services, HEALTHCOM 2011* **2011**, pp. 150–156.
29. Cherdantseva, Y.; Hilton, J. A reference model of information assurance & security. Proceedings - 2013 International Conference on Availability, Reliability and Security, ARES 2013, 2013, pp. 546–555. doi:10.1109/ARES.2013.72.
30. Aleisa, N.; Renaud, K. Privacy of the Internet of Things: A Systematic Literature Review. arXiv preprint arXiv:1611.03340, 2017, pp. 1–10. doi:10.24251/HICSS.2017.717.
31. Yu, S.; Guo, S. *Big Data Concepts, Theories, and Applications*; Springer International Publishing: Cham, 2016; pp. 1–437. doi:10.1007/978-3-319-27763-9.
32. Grobauer, B.; Walloschek, T.; Stöcker, E. Understanding cloud computing vulnerabilities. *IEEE Security and Privacy* **2011**, *9*, 50–57. doi:10.1109/MSP.2010.115.
33. OWASP. The Ten Most Critical Web Application Security Risks. Technical report, OWASP, 2010.
34. Harnik, D.; Pinkas, B.; Shulman-Peleg, A. Side Channels in Cloud Services: Deduplication in Cloud Storage. *IEEE Security & Privacy Magazine* **2010**, *8*, 40–47. doi:10.1109/MSP.2010.187.
35. Masdari, M.; Jalali, M. A survey and taxonomy of DoS attacks in cloud computing. *Security and Communication Networks* **2016**, *9*, 3724–3751. doi:10.1002/sec.1539.
36. IBM-Security. 2016 Cyber Security Intelligence Index. Technical report, IBM, 2016.
37. EY. Managing insider threat A holistic approach to dealing with risk from within. Technical report, EY, 2015.
38. Kaaniche, N.; Laurent, M. Data security and privacy preservation in cloud storage environments based on cryptographic mechanisms, 2017. doi:10.1016/j.comcom.2017.07.006.
39. Kaaniche, N. Cloud data storage security based on cryptographic mechanisms. PhD thesis, Institut National des Télécommunications, 2014.
40. Rittinghouse, J.; Ransome, J. *Cloud computing Implementation, Management, and Security*; CRC press, 2010; p. 340.

41. Stevens, M.; Lenstra, A.; de Weger, B. Chosen-Prefix Collisions for MD5 and Colliding X.509 Certificates for Different Identities. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*; Springer, 2007; pp. 1–22. doi:10.1007/978-3-540-72540-4{_}1.
42. Daum, M.; Lucks, S. Hash Collisions (The Poisoned Message Attack). *Eurocrypt 2005 rump session*.
43. Rashid, F.; Miri, A.; Woungang, I. A secure data deduplication framework for cloud environments. 2012 Tenth Annual International Conference on Privacy, Security and Trust. IEEE, 2012, pp. 81–87. doi:10.1109/PST.2012.6297923.
44. Yan, Z.; Wang, M.; Li, Y.; Vasilakos, A.V. Encrypted Data Management with Deduplication in Cloud Computing. *IEEE Cloud Computing* **2016**, *3*, 28–35. doi:10.1109/MCC.2016.29.
45. Puzio, P.; Molva, R.; Onen, M.; Loureiro, S. ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage. 2013 IEEE 5th International Conference on Cloud Computing Technology and Science. IEEE, 2013, pp. 363–370. doi:10.1109/CloudCom.2013.54.
46. Xu, J.; Chang, E.C.; Zhou, J. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security - ASIA CCS '13; ACM Press: New York, New York, USA, 2013; p. 195. doi:10.1145/2484313.2484340.
47. Shin, Y.; Koo, D.; Hur, J. A Survey of Secure Data Deduplication Schemes for Cloud Storage Systems. *ACM Computing Surveys* **2017**, *49*, 1–38. doi:10.1145/3017428.
48. Jiang, H.; Shen, F.; Chen, S.; Li, K.C.; Jeong, Y.S. A secure and scalable storage system for aggregate data in IoT. *Future Generation Computer Systems* **2015**, *49*.
49. Kumar, A.; Lee, B.G.; Lee, H.; Kumari, A. Secure storage and access of data in cloud computing. 2012 International Conference on ICT Convergence (ICTC). IEEE, 2012, pp. 336–339. doi:10.1109/ICTC.2012.6386854.
50. Bokefode, J.D.; Bhise, A.S.; Satarkar, P.A.; Modani, D.G. Developing A Secure Cloud Storage System for Storing IoT Data by Applying Role Based Encryption. *Procedia Computer Science* **2016**.
51. Fu, J.S.; Liu, Y.; Chao, H.C.; Bhargava, B.K.; Zhang, Z.J. Secure Data Storage and Searching for Industrial IoT by Integrating Fog Computing and Cloud Computing. *IEEE Transactions on Industrial Informatics* **2018**, *14*, 4519–4528. doi:10.1109/TII.2018.2793350.
52. Fu, Z.; Cao, X.; Wang, J.; Sun, X. Secure storage of data in cloud computing. Proceedings - 2014 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2014. IEEE, 2014, pp. 783–786. doi:10.1109/IIH-MSP.2014.199.
53. Rao, B.T.; others. A study on data storage security issues in cloud computing. *Procedia Computer Science* **2016**, *92*, 128–135.
54. Liu, H.; Wang, H.; Chen, Y. Ensuring data storage security against frequency-based attacks in wireless networks. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, Vol. 6131 LNCS, pp. 201–215. doi:10.1007/978-3-642-13651-1{_}15.
55. Storer, M.W.; Greenan, K.M.; Miller, E.L.; Voruganti, K. POTSHARDS: Secure Long-Term Storage Without Encryption. *ATC* **2007**, pp. 143–156.
56. Jayant.D, B.; Swapnaja A, U.; Sulabha S, A.; Dattatray G, M. Analysis of DAC MAC RBAC Access Control based Models for Security. *International Journal of Computer Applications* **2014**, *104*, 6–13. doi:10.5120/18196-9115.
57. Wang, J.K.; Jia, X. Data security and authentication in hybrid cloud computing model. 2012 IEEE Global High Tech Congress on Electronics. IEEE, 2012.
58. Sandhu, R.; Coyne, E.; Feinstein, H.; Youman, C. Role-based access control models. *Computer* **1996**, *29*, 38–47.
59. Sandhu, R.; Bhamidipati, V. The ASCAA principles for next-generation role-based access control. *ARES 2008 - 3rd International Conference on Availability, Security, and Reliability, Proceedings* **2008**. doi:10.1109/ARES.2008.211.
60. Xiao, M.; Zhou, J.; Liu, X.; Jiang, M. A hybrid scheme for fine-grained search and access authorization in fog computing environment. *Sensors (Switzerland)* **2017**, *17*, 1–22.
61. Zuo, C.; Shao, J.; Wei, G.; Xie, M.; Ji, M. CCA-secure ABE with outsourced decryption for fog computing. *Future Generation Computer Systems* **2016**, *78*, 730–738. doi:10.1016/j.future.2016.10.028.

62. Jiang, Y.; Susilo, W.; Mu, Y.; Guo, F. Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing. *Future Generation Computer Systems* **2018**, *78*, 720–729.
63. Yu, Z.; Au, M.H.; Xu, Q.; Yang, R.; Han, J. Towards leakage-resilient fine-grained access control in fog computing. *Future Generation Computer Systems* **2018**, *78*, 763–777.
64. Abdelwahab, S.; Hamdaoui, B.; Guizani, M.; Znati, T. Replisom: Disciplined Tiny Memory Replication for Massive IoT Devices in LTE Edge Cloud. *IEEE Internet of Things Journal* **2016**, *3*, 327–338. doi:10.1109/JIOT.2015.2497263.
65. Al-Arnaout, Z.; Fu, Q.; Frean, M. A divide-and-conquer approach for content replication in WMNs. *Computer Networks* **2013**, *57*, 3914–3928. doi:10.1016/j.comnet.2013.09.016.
66. Al-Arnaout, Z.; Fu, Q.; Frean, M. Exploiting graph partitioning for hierarchical replica placement in WMNs. Proceedings of the 16th ACM international conference on Modeling, analysis & simulation of wireless and mobile systems - MSWiM '13; ACM Press: New York, New York, USA, 2013; pp. 5–14. doi:10.1145/2507924.2507928.
67. Zhang, Q.; Zhang, S.Q.; Leon-Garcia, A.; Boutaba, R. Aurora: Adaptive Block Replication in Distributed File Systems. 2015 IEEE 35th International Conference on Distributed Computing Systems. IEEE, 2015, pp. 442–451. doi:10.1109/ICDCS.2015.52.
68. Riahi Sfar, A.; Natalizio, E.; Challal, Y.; Chtourou, Z. A roadmap for security challenges in the Internet of Things. *Digital Communications and Networks* **2018**, *4*, 118–137. doi:10.1016/j.dcan.2017.04.003.
69. W., L.; B., F.; L., Y.; X., Y. A tree based location privacy approach against multi-precision continuous attacks in the internet of things. *Journal of Information and Computational Science* **2012**, *9*, 1807–1819.
70. Xu, Y.; Qin, X.; Yang, Z.; Yang, Y.; Huang, C. An algorithm of k-anonymity for data releasing based on fine-grained generalization. *Journal of Information and Computational Science* **2012**, *9*.
71. Machanavajjhala, A.; Kifer, D.; Gehrke, J.; Venkitasubramaniam, M. Diversity : Privacy Beyond k-Anonymity. *ACM Transactions on Knowledge Discovery from Data* **2007**.
72. Li, N.; Li, T.; Venkatasubramanian, S. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. 2007 IEEE 23rd International Conference on Data Engineering. IEEE, 2007, pp. 106–115. doi:10.1109/ICDE.2007.367856.
73. Rebollo-Monedero, D.; Forné, J.; Domingo-Ferrer, J. From t-Closeness-like privacy to postrandomization via information theory. *IEEE Transactions on Knowledge and Data Engineering* **2010**.
74. Narendra, N.C.; Nayak, S.; Shukla, A. Managing large-scale transient data in IoT systems. 2018 10th International Conference on Communication Systems and Networks, COMSNETS 2018, 2018, Vol. 2018-Janua, pp. 565–568. doi:10.1109/COMSNETS.2018.8328274.
75. Cecchinel, C.; Jimenez, M.; Mosser, S.; Riveill, M. An Architecture to Support the Collection of Big Data in the Internet of Things. 2014 IEEE World Congress on Services. IEEE, 2014, pp. 442–449. doi:10.1109/SERVICES.2014.83.
76. Fazio, M.; Puliafito, A.; Villari, M. IoT4S: a new architecture to exploit sensing capabilities in smart cities. *International Journal of Web and Grid Services* **2014**, *10*, 114. doi:10.1504/IJWGS.2014.060255.
77. Narendra, N.C.; Koorapati, K.; Ujja, V. Towards Cloud-Based Decentralized Storage for Internet of Things Data. 2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM). IEEE, 2015, pp. 160–168. doi:10.1109/CCEM.2015.9.
78. Gentry, C. Fully homomorphic encryption using ideal lattices. Proceedings of the 41st annual ACM symposium on Symposium on theory of computing - STOC '09. ACM, 2009, p. 169. doi:10.1145/1536414.1536440.
79. Curtmola, R.; Garay, J.; Kamara, S.; Ostrovsky, R. Searchable symmetric encryption : Improved definitions. *Journal of Computer Security* **2011**, *19*, 895–934. doi:10.3233/JCS-2011-0426.
80. Wang, P.; Wang, H.; Pieprzyk, J. Threshold Privacy Preserving Keyword Searches. In *SOFSEM 2008: Theory and Practice of Computer Science*; Springer Berlin Heidelberg: Berlin, Heidelberg, 2008. doi:10.1007/978-3-540-77566-9{_}56.
81. Wang, P.; Wang, H.; Pieprzyk, J. An efficient scheme of common secure indices for conjunctive keyword-based retrieval on encrypted data. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer, Berlin, Heidelberg, 2009, Vol. 5379 LNCS, pp. 145–159. doi:10.1007/978-3-642-00306-6{_}11.

82. Yang, Y.; Lu, H.; Weng, J. Multi-User Private Keyword Search for Cloud Computing. 2011 IEEE Third International Conference on Cloud Computing Technology and Science. IEEE, 2011.
83. Cheung, L.; Newport, C. Provably secure ciphertext policy ABE. Proceedings of the 14th ACM conference on Computer and communications security - CCS '07; ACM Press: New York, New York, USA, 2007.
84. Sun, W.; Yu, S.; Lou, W.; Hou, Y.T.; Li, H. Protecting Your Right: Verifiable Attribute-Based Keyword Search with Fine-Grained Owner-Enforced Search Authorization in the Cloud. *IEEE Transactions on Parallel and Distributed Systems* **2016**, *27*.
85. Sun, W.H.; Yu, S.C.; Lou, W.J.; Hou, Y.T.; Li, H.; Ieee. Protecting Your Right: Attribute-based Keyword Search with Fine-grained Owner-enforced Search Authorization in the Cloud. In *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*; IEEE, 2014; pp. 226–234.
86. Shu, J.; Shen, Z.; Xue, W. Shield: A stackable secure storage system for file sharing in public storage. *Journal of Parallel and Distributed Computing* **2014**, *74*, 2872–2883. doi:10.1016/j.jpdc.2014.06.003.
87. Ambade, A.D.; Pansare, J.R. Securing Data Storage System for Internet of Things Using Key Aggregate Cryptosystem. *International Journal of Scientific & Engineering Research* **2017**, *8*, 31.
88. Adluru, P.; Datla, S.S.; Zhang, X. Hadoop eco system for big data security and privacy. 2015 Long Island Systems, Applications and Technology. IEEE, 2015, pp. 1–6. doi:10.1109/LISAT.2015.7160211.
89. Saraladevi, B.; Pazhaniraja, N.; Paul, P.V.; Basha, M.S.; Dhavachelvan, P. Big Data and Hadoop-a Study in Security Perspective. *Procedia Computer Science* **2015**, *50*, 596–601. doi:10.1016/J.PROCS.2015.04.091.
90. Huang, Z.; Su, X.; Zhang, Y.; Shi, C.; Zhang, H.; Xie, L. A decentralized solution for IoT data trusted exchange based-on blockchain. *2017 3rd IEEE International Conference on Computer and Communications, ICCC 2017* **2018**, *2018-Janua*, 1180–1184. doi:10.1109/CompComm.2017.8322729.
91. Shafagh, H.; Burkhalter, L.; Hithnawi, A.; Duquennoy, S. Towards Blockchain-based Auditable Storage and Sharing of IoT Data. Proceedings of the 2017 on Cloud Computing Security Workshop. ACM, 2017, pp. 45–50.
92. Xu, Q.; Aung, K.M.M.; Zhu, Y.; Yong, K.L. A Blockchain-Based Storage System for Data Analytics in the Internet of Things. In *Studies in Computational Intelligence*; Springer, 2018; Vol. 715, pp. 119–138. doi:10.1007/978-3-319-58190-3{_}8.
93. Gholami, A.; Laure, E. Big Data Security and Privacy Issues in the CLOUD. *International Journal of Network Security & Its Applications* **2016**, *8*, 59–79. doi:10.5121/ijnsa.2016.8104.
94. Anand, M. Cloud Monitor: Monitoring Applications in Cloud. 2012 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM). IEEE, 2012, pp. 1–4. doi:10.1109/CCEM.2012.6354603.
95. Brinkmann, A.; Fiehe, C.; Litvina, A.; Luck, I.; Nagel, L.; Narayanan, K.; Ostermair, F.; Thronicke, W. Scalable Monitoring System for Clouds. 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing. IEEE, 2013, pp. 351–356.
96. Nikolai, J.; Yong Wang. Hypervisor-based cloud intrusion detection system. 2014 International Conference on Computing, Networking and Communications (ICNC). IEEE, 2014, pp. 989–993. doi:10.1109/ICCNC.2014.6785472.
97. Marchal, S.; Jiang, X.; State, R.; Engel, T. A Big Data Architecture for Large Scale Security Monitoring. 2014 IEEE International Congress on Big Data. IEEE, 2014, pp. 56–63. doi:10.1109/BigData.Congress.2014.18.
98. Liu, C.; Ranjan, R.; Yang, C.; Zhang, X.; Wang, L.; Chen, J. MuR-DPA: Top-Down Levelled Multi-Replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud. *IEEE Transactions on Computers* **2015**, *64*, 2609–2622. doi:10.1109/TC.2014.2375190.
99. Alliance, A.S.C. Embedded Hardware Security for IoT Applications. *A Smart Card Alliance Internet of Things Security Council White Paper* **2016**.
100. Sushma, M.; Jaidhar, C.D.; Gudisagar, C.; Sahoo, B.R. Secure data migration between cloud storage systems. *2017 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2017* **2017**, *2017-Janua*, 2208–2212. doi:10.1109/ICACCI.2017.8126173.
101. Shen, Q.; Zhang, L.; Yang, X.; Yang, Y.; Wu, Z.; Zhang, Y. SecDM: Securing Data Migration between Cloud Storage Systems. 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing. IEEE, 2011, pp. 636–641.
102. Dhamija, A.; Dhaka, V. A novel cryptographic and steganographic approach for secure cloud data migration. 2015 International Conference on Green Computing and Internet of Things (ICGCIoT). IEEE, 2015, pp. 346–351.

103. Khalil, I.; Hababeh, I.; Khreishah, A. Secure inter cloud data migration. 2016 7th International Conference on Information and Communication Systems (ICICS). IEEE, 2016, pp. 62–67.
104. S. S., M.; S., R. Security Architecture for multi-Tenant Cloud Migration. *International Journal of Future Computer and Communication* **2018**, *7*, 42–45. doi:10.18178/ijfcc.2018.7.2.518.
105. Alok Kumbhare, Yogesh Simmhan, V.P. Designing a Secure Storage Repository for Sharing Scientific Datasets using Public Clouds. *DataCloud-SC '11 Proceedings of the second international workshop on Data intensive computing in the clouds* **2011**, pp. 31–40.
106. European Parliament and Council of the European Union. General Data Protection Regulation (GDPR) – Final text neatly arranged.
107. Spiekermann, S.; Cranor, L.F. Engineering privacy. *IEEE Transactions on Software Engineering* **2009**, *35*, 67–82. doi:10.1109/TSE.2008.88.
108. Spiekermann, S.; Cranor, L.F. Engineering Privacy on Software Engineering ,. *IEEE Transactions on Software Engineering* **2009**, *35*, 67–82.
109. Hoepman, J.H. Privacy Design Strategies. In *IFIP International Information Security Conference*; Springer, 2014; pp. 446–459. doi:10.1007/978-3-642-55415-5{_}38.
110. OWASP_Foundation. Security by Design Principles, 2016.
111. Kotzanikolaou, P. Data retention and privacy in electronic communications. *IEEE Security and Privacy* **2008**, *6*, 46–52. doi:10.1109/MSP.2008.114.
112. Xu, Z.; Martin, K.; Kotnik, C.L. A Survey of Security Services and Techniques in Distributed Storage Systems. Technical report, The Steering Committee of The World Congress in Computer Science, Computer˜. . . , 2010.
113. PICDSS. Requirements and Security Assessment Procedures Document Changes. Technical report, PCI Security Standards Council, 2016.
114. Beynon-Dames, P. Database and expert systems applications. *Engineering Applications of Artificial Intelligence* **1996**, *9*, 575. doi:10.1016/0952-1976(96)84165-0.
115. Ma, Y.; Guo, Y.; Tian, X.; Ghanem, M. Distributed Clustering-Based Aggregation Algorithm for Spatial Correlated Sensor Networks. *IEEE Sensors Journal* **2011**, *11*, 641–648. doi:10.1109/JSEN.2010.2056916.
116. Lindsey, S.; Raghavendra, C.; Sivalingam, K.M. Data gathering algorithms in sensor networks using energy metrics [PEGASIS]. *IEEE Transactions on Parallel and Distributed Systems* **2002**, *13*, 924–935.
117. Danezis, G.; Domingo-Ferrer, J.; Hansen, M.; Hoepman, J.H.; Le Métayer, D.; Tirtea, R.; Schiffner, S. Privacy and Data Protection by Design – from policy to engineering. *ENISA* **2014**.
118. Vanitha, M.; Kavitha, C. Secured data destruction in cloud based multi-tenant database architecture. *2014 International Conference on Computer Communication and Informatics: Ushering in Technologies of Tomorrow, Today, ICCCI 2014* **2014**, pp. 1–6. doi:10.1109/ICCCI.2014.6921774.
119. Han, J.; Pei, J.; Kamber, M. *Data mining: concepts and techniques*; Elsevier, 2011.
120. Weingart, S.H. Physical Security Devices for Computer Subsystems: A Survey of Attacks and Defenses. *springer* **2000**, pp. 302–317. doi:10.1007/3-540-44499-8{_}24.
121. Terzi, D.S.; Terzi, R.; Sagiroglu, S. A survey on security and privacy issues in big data. 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST). IEEE, 2015, pp. 202–207. doi:10.1109/ICITST.2015.7412089.
122. Luan, T.H.; Cai, L.X.; Chen, J.; Shen, X.S.; Bai, F. Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks. *IEEE Transactions on Vehicular Technology* **2014**, *63*, 1419–1435. doi:10.1109/TVT.2013.2251924.
123. Department of Homeland Security (DHS). Strategic Principles for Securing the IoT (version 1.0). Technical Report November, U.S. Department of Homeland Security, 2016.
124. Cloud Standards Customer Council. Security for Cloud Computing 10 Steps to Ensure Success, 2015.
125. Mungole, A.J.; Dhore, M.P. Techniques of Data Migration in Cloud Computing. *IEEE ACCESS* **2016**, *36*, 36–38.
126. Kushwah, V.S. A Security approach for Data Migration in Cloud Computing. *International Journal of Scientific and Research Publications* **2013**, *3*, 1–8.
127. Kumar, P.R.; Raj, P.H.; Jelciana, P. Exploring Data Security Issues and Solutions in Cloud Computing. *Procedia Computer Science* **2018**, *125*, 691–697. doi:10.1016/j.procs.2017.12.089.

128. Claycomb, W.R.; Nicoll, A. Insider threats to cloud computing: Directions for new research challenges. *Proceedings - International Computer Software and Applications Conference* **2012**, pp. 387–394. doi:10.1109/COMPSAC.2012.113.
129. Sallam, A.; Bertino, E.; Hussain, S.R.; Landers, D.; Lefler, R.M.; Steiner, D. DBSAFE—An Anomaly Detection System to Protect Databases From Exfiltration Attempts. *IEEE Systems Journal* **2017**, *11*, 483–493.
130. Federal Trade Commission. IoT Privacy & Security in a Connected World. Technical Report January, Federal Trade Commission, 2015.
131. Shan, C.; Heng, C.; Xianjun, Z. Inter-cloud operations via NGSON. *IEEE Communications Magazine* **2012**, *50*, 82–89.
132. Sotiriadis, S.; Bessis, N.; Petrakis, E.G.M. An inter-cloud architecture for future internet infrastructures. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2014**, *8907*, 206–216.
133. Borylo, P. Intercloud: Solving Interoperability and Communication in a Cloud of Clouds (Frahim, J., et al; 2016) [Book Review]. *IEEE Communications Magazine* **2017**, *55*, 6–6. doi:10.1109/mcom.2017.7876847.
134. Cisco. The Internet of Things Reference Model. *Internet of Things World Forum* **2014**, pp. 1–12.
135. Venčkauskas, A.; Morkevicius, N.; Bagdonas, K.; Damaševičius, R.; Maskeliūnas, R. A lightweight protocol for secure video streaming. *Sensors (Switzerland)* **2018**, *18*. doi:10.3390/s18051554.

Sample Availability: Samples of the compounds ..... are available from the authors.