Health Wearables and Privacy

L. Antelo Blanco

***Abstract*—Health wearables provide a great tool for people following the quantified self-movement. These devices are able to measure tons of metrics which are usually stored in the cloud. Over time privacy concerns have risen to such an extent that the GDPR was released in order to appease these concerns. In this report, we will investigate the current situation as well as the customer's privacy concerns with respect to the collected data. Furthermore, we will have a look at different strategies how privacy may be protected. Finally, we will analyze Pilatus, that uses partially homomorphic encryption for storing and sharing data among users. Pilatus provides a good ad-hoc solution that fulfills many constraints set by the GDPR and approaches the current privacy concerns. In addition, Pilatus can seamlessly be integrated into the current IT infrastructure, that health wearable manufactures and service providers use to serve their customers.**

1. Introduction

As of May 25 2018, the General Data Protection Regulation (GDPR) [8] has come into effect and for all companies operating in the European Union this imposes a new set of rules empowering their citizens to have more control over their data. This set of rules also applies to companies like Apple, Fitbit and others. This newly imposed regulation shows, that the privacy concerns have reached a limit, where the European government was forced to act. This new regulation was created in 2016 and is now officially legally binding. Privacy concerns are usually stated when dealing with collections, processing of data and data sharing. Unfortunately, there is no in depth research concerning the question what data privacy actually means and what the consequences are. Furthermore the answer to the question what kind of data privacy concerns the population might have, has never actually been investigated. Nevertheless, concerns have been expressed by many forums, researchers, newspapers and friends. In 2018, M. Becker performed a structured investigation figuring out the possible thoughts on data privacy and concluded that there are three highly problematic and symptomatic points that the participants frequently mentioned like

- Dilemma of Forced Acceptance,
- State-Trait Data Sensitivity and
- Transparency [18].

On the other hand, we should not forget, that these companies are in a highly competitive market where many are trying to sell their products and their services. These companies rely on the data in order to improve their services and products. Therefore, we have to ask ourselves, how we can establish a fair solution for both sides with the goal to preserve users' privacy over their data and to allow the companies to run analyses over this data such that personalized feedback may be provided or insights gained. Over the last decades progression has been made in the field of cryptography, such that data can be securely encrypted and stored. Nowadays more and more of this data is stored in the cloud. On these cloud servers analyses over the data may be performed by the user and cloud service providers as well. Thanks to the advances of homomorphic encryption users can restrict the access to their data and still perform analyses without disclosing any information [21]. The advances in the field of machine learning and artificial intelligence allow to gain insights and create new services that are based on data, which is still possible when applying homomorphic encryption to run statistical analyses over encrypted data [30]–[32], [35] and moreover to use this encrypted data to train models [33].

This report is divided into 6 sections. In Section 2 we will give an overview over the data, that wearables are able to generate. This data gives companies the advantage to run analyses, gain insights and improve their product and services. In Section 3 we will look at the results by M. Becker [18], where he investigated the data privacy concerns that users may have and will form the basis for the following sections. In Section 4 we will discuss possible solutions based on the main data privacy concerns. In section 5 we will focus our attention to a strategy that seems to be easily implementable. This solution is named Pilatus [31] and allows to encrypt and store data in the cloud, that was generated by health wearables. Furthermore Pilatus allows to share encrypted data with particular users or within a group without having to decrypt the data at any point in the cloud. In section 6 we will state our personal conclusion based on this accumulated knowledge.

2. Quantified-Self Data

The quantified self movement having its beginning in 2007 [14] has gained a lot of followers and wherever we look, we always find people having a wearable on its wrist measuring tons of metrics like steps, activity level and if available also heart rate. Usually the manufacturers provide software in order to update, maintain and retrieve the data generated by these health wearables. Depending on the implementation of the communication protocols the data will be sent using an end-to-end encrypted channel to the cloud, where it will be stored. This data can be read and analyzed by these service providers, who gain insights and are able to improve their product and services. This allows service provides to give moreover personal feedback as it is the case for the Fitbit Versa [16].

In this section we want to give you an overview over the range of parameters wearables are able to measure and that manufactures are able to collect. We will focus the three most common wearable types based on their operating system or their trademark. This allows us to have an overview over the scale of vital parameters and indirect vital parameters that are measurable and analyzable.

2.1. Apple Watch Series 3

The Apple Watch Series 3 [1] was released in September 2017 and was improved in several ways compared to its predecessor. This wearable has a respectable computation power for its size and is fully packed with a lot of sensors capable of generating a huge amount of data per time frame. This is only limited by its battery life and memory constraint. The Apple Watch Series 3 implements

- **GPS**, **GLONASS**, **Galileo**, and **QZSS**,
- a **barometric altimeter**,
- a **heart rate sensor**,
- an **accelerometer**,
- a **gyroscope** and
- an **ambient light sensor**.

The collection of the data generated by these sensors is done with or without any awareness of the Apple Watch users. This data is usually backed up on an iphone [3] through which it may be stored in the cloud or transmitted to other appropriate service providers with the users' consent.

2.2. Fitbit Versa

Fitbit Versa is a new product line by the health wearable producer Fitbit [11]. It was released in 2018 with women as customers in mind. This health wearable tracks vital parameters and further metrics. Based on the data Fitbit provides personalized feedback, which is only possible when being able to process and analyze the data [16]. Otherwise no personalized feedback can be generated. The Fitbit Versa provides

- a **3-axis accelerometer**,
- a **3-axis gyroscope**.
- an **optical heart rate monitor**,
- a **SpO2 sensor**,
- an **altimeter** and
- an **ambient light sensor** [9]

on which Fitbit makes its inferences on how it can best help its customer with feedback and recommendations in order to improve their health and fitness. This data is usually sent to and stored in the cloud. The location of these servers may differ from the users location and the degree of encryption and security is unknown to my best knowledge.

2.3. Android Based Wearable

This group consist of a vast variety of different manufactures that base their device's operation system on Android or use an Android application to be functional, except the aforementioned two wearable lines. We provide here a summary of the available sensors by wearables mentioned in one review of Android Authority [4]. We can expect to find

- **GPS**,
- a **3-axis accelerometer**,
- a **3-axis gyroscope**,
- an **optical heart rate monitor**,
- an **altimeter** and
- an **ambient light sensor**

among others. The data collected by these sensor usually need to be offloaded onto a coupled device due to the limited memory available. The offloaded data may furthermore be transmitted to the manufacturer or stored on one of their websites and thus in the cloud. Some manufactures do not allow to use their device without having first installed a software or downloaded an application, which needs to sync with their wearable. Moreover some manufacturers go further and explicitly ask for a registration. Without these steps the use of these wearables may be not possible like it is the case for the Polar A360. This wearable is not usable without having it connected to one external device that runs a Polar software and therefore forcing users to register their products in order to be fully functional.

3. Data Privacy Concerns

The high data sensitivity as well as the mobility of personal electronic devices show that privacy concerns are more important in the context of health wearables. These health wearables usually store the user's personal health information in the cloud, what allows the provider to gain more insights about their users. Moritz Becker investigated in his paper [18] the driving factors of health information privacy concerns and the underlying rational of users' privacy perception with respect to commercially available health wearables without the need of healthcare professionals. By using an iterative thematic analysis approach health wearables users were assigned to seven semi-structured focus groups with six users of health wearables each, where the participants discussed health wearable and privacy related topics given by a thematic map. This investigative method is based on the Health Information Privacy Concerns Model composed of six dimensions to explicitly address privacy concerns with health information technologies [23]. We give a short overview over these six dimensions and review the results by M. Becker for each dimension separately.

- **Collection**
  describes the subjective concern with respect to the accumulation and storage of personal health related information.
- **Unauthorized Secondary Use**
  addresses the concerns over the use of users' data other than agreed upon.
- **Improper Access**
  describes users' concerns over the perceived threat of unauthorized access by third parties.
- **Errors**
  describes the data inaccuracies and its consequences for the respective health wearable users.
- **Control**
  covers the user's perceived control over the personal health information.
- **Awareness**
  addresses the concerns regarding the lack of awareness

how personal health information is used and moreover protected.

3.1. Collection

Participants discussed their concerns over the degree of anonymization of their personal health information and some were very concerned. Their personal health information can provide some clues participants do not want to share with others. Furthermore they were afraid that by combining databases from different sources anonymization might be worthless and would consequently reveal further information participants wish to disclose.

In addition, the location of data storage and analysis shows to have an impact how participants express their concerns. Nevertheless, they value the collection of personal health information as an asset in order to share them with their healthcare providers or simply by showing others and get feedback.

3.2. Unauthorized Secondary Use

Participants fear their data might be used for other purposes than agreed upon. By cross-connecting their data without their consent companies gain more insights, health wearable users might not want, and do not know if this happened, happens or will happen. Along this cross-connection errors might arise due to the analyses over this expanded data set, as devices do not always measure accurately, which might be harmful when analyses will build the base for future predictions by companies over their users.

Participants are aware that their data might be sold to other companies and hope, that their data might be securely anonymized. Unfortunately the laws used not to be so strict and painful enough such that the GDPR [6] provides a change in this regard. With this new instantiated law, companies now have to inform users over such transfers, and moreover if its data is used for predictive decision making. In addition, the companies are now only allowed to use the data for well defined purposes for customers in the European Union. Finally, users put the disclosure of their personal health information into a context when agreeing to the companies' term of use, depending if they really want to use the device or not. But it depends also on what purpose, as for product improvement as well as for medical research no concern were expressed compared to insurance companies, if the disclosure would imply a disadvantage.

3.3. Improper Access

Participants showed concern towards security gaps depending on access route between inappropriate access channels due to insufficient privacy adjustment options. Or due to the companies financial interest and lack of implementation of privacy features. The latter is caused by the fact, that providers have to invest money for privacy and can therefore not make money with privacy. In the discussion, they also showed concern toward hacking hazards, as they are afraid of being hacked and their data stolen and consequently used for purposes out of their control.

3.4. Errors

Health wearables do not always measure accurately which might lead to false conclusions when analyzing the data for an insurance application or other purposes. This data inaccuracy concerns the participants whenever this leads to economic and social disadvantages. In case data would be less inaccurate participants would be less concerned with privacy issues. Participants also showed concerns with further consequences due to disclosure of their personal health information which might cause them to be billed more expensive insurance options or when healthcare professionals use the data to decide upon clinical treatments. Hence, depending on the anticipated consequence participants showed differentiated opinions.

Finally the error types and their effects were discussed within the groups as the collected data might provide false conclusions from either side and provoke unfortunate actions causing physical or economic harm on their users.

3.5. Control

Participant do not have an overview over the extent of data collection nor control over the analyses run over this data, which the GDPR solves by imposing transparency over the business approach on this data.

The dilemma of forced acceptance remains a very concerning topic to the participants as when using a health wearable users are forced to agree on such terms in order to use the devices. Hence, a full control over the collected data is not possible and a refusal using these devices would look like a step backward, which the GDPR does not solve so far. A participant stated that Ï would like to to have more control about my data, but you have to accept how it goes. Beggars cannot be chooserswhich sums up the extent of discrepancy a user forgoes when accepting the terms of use. As a consequence, by purchasing a product and accepting a product, the collected data is locked and furthermore the terms of use are usually updated leaving the user with no choice as to accept them. Otherwise they loose their data, their device they invested in as they cannot use it anymore. If you want to continue to use the device you have to accept it or you loose. Hence you are locked in their cage.

3.6. Awareness

Participant discussed concerns with respect to state-trait data sensitivity as depending on the nature of the data collection this data is regarded as highly sensitive, the more personal it gets. Hence it depends on the device's focus. When it is medically focused, participants were more concerned compared to devices with a fitness and lifestyle focus, though it depended on the users relative perspective over such data.

M. Becker spotted that participants expressed their helplessness towards such data collections and practices through trivializations showing signs of resignation towards this problem. Finally, participants feared that it will get worse and that future generations might not care at all. Therefore the limit with regard to privacy will be stretched further and further due to the indifference towards such problems leaving future customers no privacy at all. Thanks to the GDPR a limit was set, which will be difficult to stretch in the coming years and marks an example for other countries as well.

3.7. Main concerns

Along the discussions some topics stood out that caught the attention of the observer, who conducted the studies. These topics stood out by their discussion's length as well as their frequency. M. Becker concluded that

- **Dilemma of Forced Acceptance**,
- **State-Trait Data Sensitivity** and
- **Transparency**

formed the three main points of concern. Transparency seems to be the only of these three, which is actually addressed by the GDPR. The findings of M. Becker will provide us with a framework when discussing the strategies.

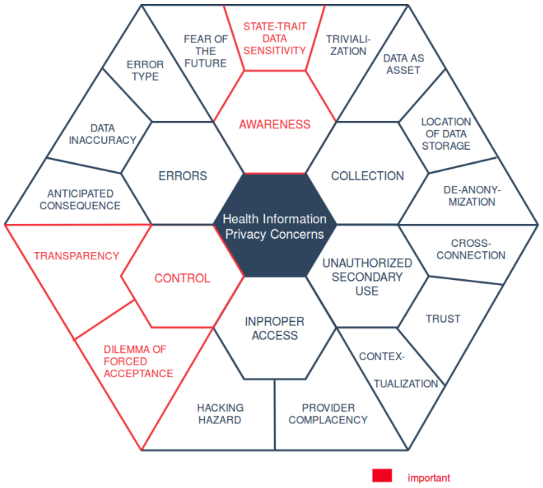

Fig. 1. Results by M. Becker.

4. Strategies

M. Becker study has shown us what kind of privacy concerns users are dealing with when using health wearables. With the GDPR into effect, businesses already have to comply with these new set of rules. The GDPR already addresses a wide spectrum of privacy concerns but others may not be addressed at all. Nevertheless, users might still want to have further privacy protection that might have an economic advantage for the company when providing supplementary protection mechanisms. There are several different solutions that could help businesses to comply with the GDPR and furthermore even provide new business opportunities that are not yet forseeable. In addition, these alternative solutions might provide a marketing advantage in conjunction with actual and relevant topics. These topics might be blockchain technology, artificial intelligence or anything else that is positively associated among the users. And some of them even provide a very interesting and efficient solution set. Thus, these approaches may provide a more lucrative solution from the company's and user's perspective than the status quo making them worthwhile to be explored and finally making money with privacy.

4.1. Distributed Ledger Technologies

At Cebit 2018 in Hanover, Germany, some topics were quite fascinating and even ignited businessmen and visitors alike when pronouncing these magic words. These topics were mainly artificial intelligence, internet of things and blockchain technology. Many representatives of companies with whom I spoke along this exposition already implement blockchain technology and artificial intelligence. These implementations allow them to gain an economic advantage over their competitors and moreover improving the efficiency of management processes.The blockchain technology is mainly associated with cryptocurrencies making it difficult to see the opportunities that this technology may provide. Bitcoin is based on a technology that is more than 10 years old and started to show some signs of weaknesses and flaws. This is why new technologies like Nano [26], Taraxa [15] and Iota [29], which are based on improved implementations of the blockchain technology, are trying to enter the market and already compete with each other in order to gain market shares around the world for the new era of the internet of things (IoT).
Distributed ledger technologies basically maintain a globally shared data structure on which transactions are enlisted and new transactions are appended by applying a concise algorithm among mutually untrusted participants. The main advantage of DLT lies in the fact, that the ledgers are distributed and decentralized, immutable, and resistant to censorship. In the ten year old blockchain technology the transactions are linked blocks forming a single chain, that uses a cryptographic hash function for appending a new block.

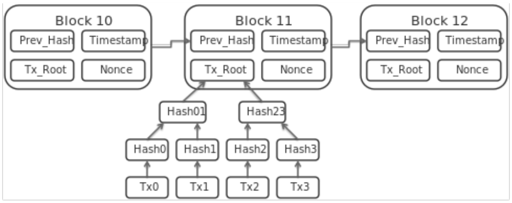

Fig. 2. A block chain segment. Image created by Matthäus Wander [5]

The new technologies try a different approach. In Iota a directed acyclic graph enlists transactions, which are entangled together, where a transaction is represented by a node in the graph. This technology is based on Tangle [29] and allows to perform fee-free and especially faster transactions compared to Bitcoin.

A different new blockchain technology forming the competitor of Iota is called Nano, where the transactions are

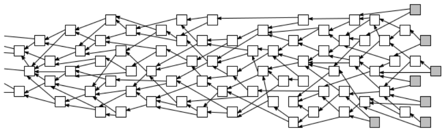

Fig. 3. An example of a ledger in Iota. Figure from the paper [29]

organized in a block-lattice and each account will be granted an account chain by its creation. This allows to give the whole ledger a structure on which it evolves. In Iota and Nano two transactions are needed to validate one transfer.

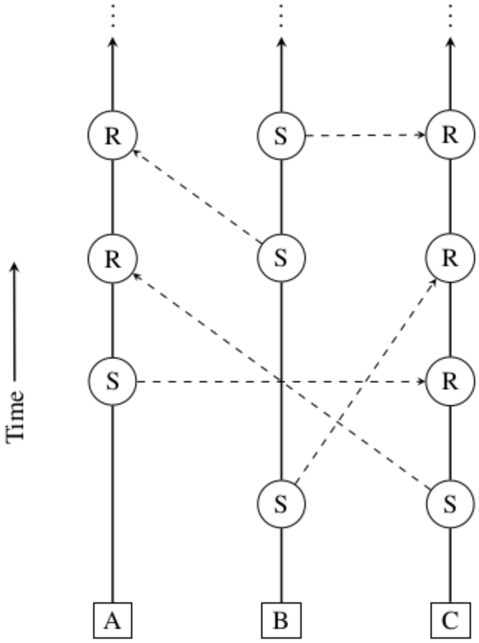

Fig. 4. An example of a ledger in Nano. Figure from the paper [26]

Iota seems to be resistant to several attack vectors and furthermore shows to have a resistance against quantum computations such that Iota seems to be a good candidate for future implementations.

The white paper [29] claims, that Iota allows a granular approach when applying data privacy. Hence users may decide more surgically to whom they want to share and what, such that subsets of data may be made public, semipublic or completely private depending on their own preferences. This is equally true for Nano and Taraxa [15], where Taraxa is actually a fork of Nano and share many properties. Theoretically, this technology may hold any kind of data [19], such that instead of financial information other kind of data can be stored and traded allowing health wearable users to store, transfer, share and sell their data in a secured way. Therefore companies may use this technology to provide a new market for their users on which they may sell their data or trade it for supplementary services.

There already exist use cases in the healthcare sector, where distributed ledger technologies are applied. One of them is the open health network [12] which provides a platform to develop mobile health applications and chatbots based on blockchain, artificial intelligence and data analytics. This allows an autonomous management of health related data for the patients and a development platform for further innovations. Moreover it unlocks further business opportunities that go far beyond privacy protection and allows business chains to perform transactions much faster enabling a faster and more reliable health care to their customers, healthcare providers and insurers.
Hence, it is recommended to explore this technology and analyze the personal needs from a manufacturer's and service provider's perspective for their health wearable customers as it may unlock more advantages than previously expected, which may cover the expenditures on privacy protection.

4.2. Information-Centric Networking

The Internet of Things comes with many challenges like security, interoperability, scalability and mobility support. The new networking paradigm information-centric networking allows us with its name-driven networking primitives to address these challenges [36]. The main concept in information-centric networking (ICN) is to make content directly addressable and therefore redundant to contact a host in order to retrieve the requested content. In other words, content is king and makes ip addresses unnecessary. Unfortunately the IoT possess inherent constraints like a small memory, limited computational capacity and power supply such that the current network protocols and architectures for ICN need to be adapted for these constraints in order to work appropriately. The IoT network architecture needs to be redesigned, such that it is scalable and secure for collecting data and monitor devices [17].
Zhang et al. propose a solution how to implement an information-centric networking architecture for sensitive data generated by health wearables [37]. Their solution allows users to decide how their health data is used and shared with other users within a well defined environment. They named their proposal NDNFit (Named Data Networking Fit) that works as a distributed mobile health application inspired by the Open mHealth ecosystem [13]. Instead of searching information or data by using an ip address, their approach allows us to search data by name, though a convenient naming convention still needs to be implemented [37]. This ICN solution provides security mechanisms by securing the content and not the communication link. The security is enforced by using a digital signature of the data's publisher for authenticating all the content. Moreover it allows for separate encryption for private content that is not intended to be shared among others. The architecture of NDNFit is based on four entities with

- a health wearable application that collects and publishes the data,
- a data storage unit for the data's persistent storage,

- a data processing unit for inferring further information around the generated data and
- a data visualization unit with a website as an interface.

The end-user can select from many different wearables, components for data storage, processing and visualization with each being from a different services providers. This advantage allows the integration and composition of many different service providers into one ecosystem.

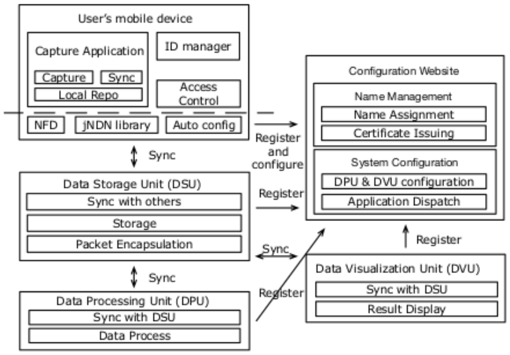

Fig. 5. Architecture of NDNFit. Figure from the paper [37]

Whenever consumers wish to retrieve data in such an ecosystem, they do this by issuing *Interests*. When checking for new data based on previous *Interests* the ecosystem provides a catalog data structure which they can fetch by issuing the appropriate *Interest* for such catalogs. The trust over the data is issued by the user who publishes the data by signing a key, that only the owner controls. These keys are stored in a data object using a standard certificate format that functions as lookup table. This trust issuing system does not prevent from fraudulent users providing misleading data generated and signed by them, though, which may lead to misleading inferences as a data consumer.

NDNFit proposes a solution for a granular access control over the data. The data owner generates a list of asymmetric key pairs, one encrypting and the other decrypting the data. The data owner encrypts the data with the encryption keys of each authorized consumer using the consumer's public key. Therefore, authorized consumers simply decrypt these decrypting keys by using their private key. This results to a key, with which data consumers can decrypt the data without having to make a connection to the data owner.

Finally it is possible to use data from different users in order to infer further information by using named functions. These functions are well described and perform all the processing of the data according to the user's or consumer's need.

Hence ICN provides a strategy to implement an ecosystem that can be compliant with the newly released GDPR and moreover the health wearable users. Moreover it allows several parties to share one ecosystem, which would avoid a potentially fractured market with different ecosystems making the flow of data a nightmare for users and service providers alike.

4.3.Homomorphic Encryption

Encrypted data usually needs to be unencrypted in order to be processed, though fully homomorphic encryption allows to process encrypted data and producing the correct value once decrypted [24]. We differentiate two homomorphic encryption schemes. One called fully homomorphic encryption (FHE), which allows to work with different operations on encrypted data, and partially homomorphic encryption (PHE), which allows only one operation to work on encrypted data like addition or multiplication over all these encrypted data points. FHE needs more computational power making it less usable for the IoT compared to PHE. Thus for health wearables and mobile phones PHE is the most suitable solution so far when using this approach, as it gives a good protection and performance with the disadvantage of not being able to perform all operations on encrypted data [24]. In addition, health wearables usually only send updates and get updates from the cloud without needing to have a lot to calculate.

In PHE the mathematical properties hold as it forms a Ring. This allows users to store their data where ever they want, allowing service providers to run analyses on this data and moreover using it to train machine learning algorithms [33] without knowing to whom data points belong to, but still need the explicit consent of the data holder in order to decrypt the results.

In order to apply PHE in the real world setting we need three entities, which consist of a client engine, a server engine and an autonomous third provider that manages the keys of the users. Users store encrypted data on not trusted servers [31]. In PHE the data is encrypted when processing it and therefore protected to the outside environment. Furthermore, it prevents malicious employees of cloud service providers to access this data. A health wearable user can simply add new data points to the cloud and process it without ever having to decrypt the data except when checking the result, for which the data is downloaded and decrypted locally. Therefore, the user does not need to decrypt the data for any manipulation (as long as it is addition or multiplication). But it cannot yet support searching [24]. Homomorphic encryption gained a lot of improvements thanks to C. Gentry who made real breakthroughs in this area. We will have a look on two implementations, namely

- **El Gamal** and
- **Paillier**,

which will be of importance when discussing a solution in section 5. Both use asymmetric public key encryption, where data is encrypted with the public key and only those having the private key associated with the public key can decrypt the data.

A. *El Gamal*

This scheme uses the hardness of solving discrete logarithms as its security strength [20]. The generation of an El Gamal key pair follows a well described algorithm which are used to encrypt (public key) and decrypt (private key) the data.

Security issues in the real world arise when the algorithm is not correctly implemented.
As it uses only multiplication on the message values it is especially suitable for multiplication over encrypted data [24]. In order to make it suitable for mobile devices, the Elliptic Curve El Gamal cryptosystem is applied. This system allows to work with smaller keys but needs to map bijectively encrypted data onto the elliptic curve in order to apply the homomorphic addition [31] on the encrypted data and not homomorphic multiplication as explained before. For more information see appendix A.

B. Paillier

This scheme uses the fact that the problem on computing the *n*th residue classes is a computational hard problem. This scheme allows for addition operations over encrypted data and the generation of a key pair follows a well described algorithm [27]. The encryption and decryption of the data follows a well described algorithm. For more information see appendix A.

PHE provides a good solution to comply with the GDPR and allow health users to control their sensitive data in the cloud. This solution is implemented by Pilatus, which will be discussed in the next section.

5. Pilatus [31]

In 2015 a team of researchers created a system that allows the secure storage of IoT data in the cloud by encrypting the data with a set of cryptographic tools such as order-preserving and PHE [32]. The name of the project was called Talos and they showed, that data from IoT devices can be efficiently encrypted and sent to the cloud. The data in the cloud would still be encrypted and thanks to to the properties of PHE manipulations can be performed over this encrypted data, without having to decrypt it. Talos uses CryptDB [28], which implements an efficient solution to query encrypted data in a database system. Unfortunately Talos did not provide any sharing functionalities such that in 2017 they extended Talos with a sharing mode and named it Pilatus. Pilatus includes further optimizations and improved the user experience compared to Talos. The team around Pilatus focused on applications that collect sensitive data generated by health wearables, especially Fitbit and Ava. For IoT devices and wearables the offload of data is necessary due to their constrained memory, which is done with the help of a mobile phone and then if preferred to the cloud.
Unfortunately the storage of data in the cloud bears risks. In the past, several hacking attacks achieved to steal data from cloud servers mitigating sensitive personal information. Furthermore companies traded user data with other companies as was the case with Facebook [34] and can not be ruled out for health wearable service providers even if they claim not to share user data with other third parties.
Pilatus allows health wearable users to encrypt and store encrypted data in the cloud servers and as well process certain queries like calculating the sum on encrypted data. Thanks to the implementation of the sharing functionality, data can be shared in an encrypted format allowing data owners to revoke access at any time by re-keying encrypted data in the cloud without ever decrypting it. Thus, the data is securely stored, shared and processed in the cloud without making the data vulnerable to confidentiality breaches.
Shafagh H. et al. provide a practical solution for mobile platforms and provide benchmarks for an implementation with Fitbit and Ava on real-world test data.
Health wearables can provide a wide range of different data consisting of vital parameters. This data is usually supplemented by activity meta-data like duration or type as well as clinical information, which mentions pain, emotions or other information.
Shafagh H. et al. claim, that Pilatus is the first solution combining support for encrypted data query processing and secure sharing. They guarantee that

1) the confidentiality of users data stored in the cloud is protected in case of compromised servers,
2) that only data of compromised users and the affected groups are not protected and
3) that it prevents unauthorized data access by providing user authentication systems

but do not guarantee the integrity of the retrieved data nor its freshness. For this purpose Pilatus can be complemented with appropriate frameworks like Verena [22].

In the following subsections, we will describe the architecture of Pilatus and the the cryptographic tools it uses. Finally we analyze the design and how Pilatus implements the functionalities over encrypted data.

5.1. Architecture

Pilatus focuses on collection, processing and sharing of sensitive data using PHE. Whenever health wearable users want to modify their encrypted data in the cloud, they can perform modifications directly in the cloud without having to download the data set and decrypt it for this purpose. We need to keep in mind, that only one set of operations may be performed either addition or multiplication on encrypted data. Pilatus is built as well as for secure sharing among users and groups. This works through re-encryption techniques. The sharing may be revoked anytime by re-keying and hence using a new key pair without having to download from or decrypt the data in the cloud.

Pilatus' architecture is built upon three components:

- **Client engine**
  Represents the user who generates, encrypts, uploads, shares and revokes the access to data. The data is generated by the health wearable, which pushes the data to the personal mobile device on which Pilatus is installed. Pilatus encrypts and decrypts this data whenever uploading or downloading data to and from the cloud.

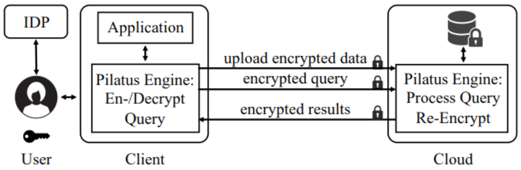

Fig. 6. The architecture of Pilatus. Figure from the paper [31].

Pilatus furthermore performs the queries and modification requests to the cloud.

- **Cloud engine**
  This componennt provides the basic database interface and features for data storage and processing by the cloud service. The cloud service is considered to be honest-but-curious and may be compromised at any time due to malicious attacks. This component runs Pilatus on top of the database management system allowing to process queries and respond to requests without ever decrypting the data stored in the cloud.
- **Identity provider**
  This component represents an independent party which verifies the user identity and public key binding. The identity provider generates the key pair and distributes it accordingly. Shafagh H. et al. [31] applied Keybase [10] as a convenient independent identity provider.

Based on these three components Pilatus can be applied for real-world applications. This solution allows modification of encrypted data, re-encryption and in-situ re-keying in the cloud and is initiated by the user running Pilatus on its mobile phone and in the cloud.

5.2. Cryptographic Tools

Pilatus uses the following tools to secure the data:

- **Partially Homomorphic Encryption (PHE)**
  PHE allows certain mathematical operations to be performed on PHE data. Pilatus makes use of the Paillier as well as the Elliptic Curve (EC) El Gamal cryptosystem to perform additive homomorphic operations on the encrypted data in the cloud. Appendix A provides a deeper look into the exact procedure as well as proof of correctness.
- **Re-Encryption**
  Whenever a user wants to share its data with another user, the sharer creates a token for the user without ever having to decrypt the data in the cloud. The sharer creates this token by using its own private key and the destination's public key. With this token the encrypted data can be transformed into still encrypted data, that can be decrypted by the destination's private key. This token is not transitive, meaning that a token from the owner to the destination, and a token from the destination to a different user, cannot create a token from owner to a different user. Therefore, the re-encrypted data for the destination, cannot be re-encrypted again for other users when applying this strategy. Hence confidentiality and security can be imposed with this approach. In order to perform the EC El Gamal re-encryption more efficiently the Chinese Remainder Theorem is applied and data packets split into smaller ones making it suitable for mobile applications and IoT. This is performed in both standard and sharing modes.
- **In-Situ Re-Keying**
  Data owner may wish to revoke shared data. This is solved by getting a new private public key pair by the IDP and encrypting the encrypted data for the new key pair by using a token sent to the cloud, which transforms the encrypted data with this token. This approach is transitive, such that the cloud service provider may reverse this approach such that this data can still be decrypted after applying the token that has been issued in the past to share the data.

5.3. Design

Along this subsection we will use an example where Alice has generated a lot of data on her Fitbit versa with Pilatus installed on her paired mobile phone as well as in the cloud run by a service provider. The data provides information about her heart frequencies and oxygen saturation along her training. Now this data will be encrypted using EC ElGamal. Pilatus allows her to run the software in **standard** or **sharing** mode. In standard mode an improved Talos is basically run, but as she wants to share her data anyway, she uses the sharing mode in order to enable cryptographically-protected sharing.

- **Processing and Sharing Encrypted Data**
  Alice has on her mobile phone all the data that was generated by her newly purchased Fitbit Versa. With Pilatus she can perform partially homomorphic encryption and upload the encrypted data in the cloud, where Pilatus is installed as well. In the cloud, Pilatus takes the encrypted data and stores it in the database provided by the cloud service provider. This data is encrypted and no information can be inferred by the cloud service provider nor by an attacker without having the appropriate keys to decrypt it or a very smart solution how to decrypt it anyway.
  She can run manipulations on the encrypted data in the cloud without having to download and decrypt it. EC El Gamal provides homomorphic addition on encrypted data and can calculate the sum if she wishes to know how much time she has already spent running this week. When she wishes to see the result of this sum, she needs to download the calculated encrypted value and decrypt it on her mobile phone.
  Bob, a good friend of her, would like to see how good she performs in her training sessions and asked her, if she could share with him this data. For this purpose Bob needs to install Pilatus on one of his personal devices and have access to the data in the cloud. After setting this up, Alice shares her data with him. For this purpose she needs his public key that was provided by the same

identity provider that generated the keys for her as well. In order to provide access to Bob, Alice needs to perform a special algorithm using a bilinear-map-based cryptosystem like AFGH by relying on the optimal Ate pairing, that creates a token with the help of her private key and Bob's public key. Her private key is not leaked when performing this algorithm nor can it be inferred from the created token. This token is used to re-encrypt the data in the cloud, she wants to share with Bobb. This token will be send by Pilatus from her mobile phone to the cloud, where the cloud sided Pilatus processes the data she wants to share with the generated token. The shared data remains encrypted but now Bob can download this data locally and decrypt it with his private key. Now he can see her data in plaintext and perform any operations for analytic purposes. It may happen, that people will ask Bob, to share her data with them. In case Bob does not care what Alice might think, he can simply share the decrypted data with them. In case Bob wants to build a token by the same special algorithm with the goal to generate a token from Alice to them and allowing them to access the shared encrypted data in the cloud directly, they will realize that it will not work as the generated tokens for the re-encryption are not transitive.

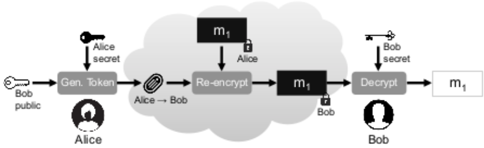

Fig. 7. Using Bob's public key and Alice's private key the token is generated to re-encrypt the data for Bob. The same mechanism is performed for group sharing. Figure from the paper [31].

- **Performance Optimization** These encryption schemes are computationally heavy for a Fitbit Versa why this task is delegated to her coupled mobile phone, but still time consuming when applying the Baby-Step-Giant-Step algorithm [2]. For this purpose the researchers came up with the idea, to disassemble the data into smaller chunks and then assemble them after performing the en-/decryption with the help of the Chinese Remainder Theorem. This allows Pilatus to reduce the computation time by a significant factor especially when decrypting the data. This approach makes Pilatus more user-friendly due to the reduced time to retrieve and decrypt data. The homomorphic properties are still valid using this approach.

- **Key Revocation**
  Now, Alice is disappointed by Bob, who shows her data to people without her permission. Therefore Alice decides to revoke his access to her data. For this purpose she needs to create a new personalized key pair by the independent identity provider. Hence she will possess a new private key and its associated public key, as well as the current private key and its associated public key. She generates a token with the same special algorithm used to generate tokens in order to share encrypted data with others, but this time to encrypt her data with the new key pair. This invalidates already issued sharing tokens used to transform encrypted data, that can be decrypted with the private key of the lucky ones . But now, the special algorithm takes the current private key and the new private key to generate the token, such that Alice can transform the encrypted data. This in-situ re-keying is performed on the server, where all the data is transformed using this token. Consequently, this data can only be decrypted with her new private key and new tokens generated with her new private key and another user's public key.
  Unfortunately this kind of transformation is transitive, such that it can be reversed by the cloud service provider who receives this token to run this transformation. This would allow access to data with old tokens by other users. Moreover, already shared data, once decrypted, may still be available in the wild. Hence it is possible that not all her personal data can be securely encrypted using this new key pair.

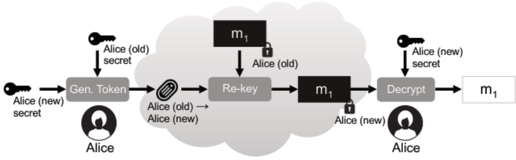

Fig. 8. Using the current and new private key the token is generated to re-encrypt the data in the cloud under the new private key. Figure from the paper [31].

- **Group Sharing Authorization**
  Alice's private data generated by her Fitbit Versa may not only be shared with a particular user but with a group of users as well. Alice creates therefore a group with a chosen name. For this group the identity provider creates a key pair. An access graph is created for this group with the node having the identity of the initiator as being Alice and the group name as well as Alice's public key. This node is signed by the identity provider and all other appended nodes by its parent node. When Alice shares her data with a group Pilatus applies an authorization mechanism to ensure that a joining group member issues the re-encryption token for the authentic group, which is performed by the parent node. After receiving the re-encryption token.
  The initiator of the group gets its ID together with the group identity signed by the IDP. The initiator signs the child nodes, and the child nodes its child nodes allowing other users to join the group by vouching for them. Now, the membership is authorized using the following two steps:

- Signing the extended identity (ID and group ID) and the public key of the new member
- Issuing a re-encryption token stored in the cloud

The re-encryption token is necessary to decrypt group data and hence to access it. The group itself possesses a key pair which is signed by the initiator. This protects the key pair being misused by a fake group having the same name. If Alice wishes to restrict new memberships, she simply needs to encrypt the group's private key with her own private key. Like this new members may only join the group with her explicit consent using a token from her to the new member in order to access the group key pair.

When someone wants to leave the group, Alice needs to start a revocation procedure. Alice therefore initiates the generation of a new personal key pair with which the group keys are re-encrypted and the access graph updated without ever exposing her private key nor those of the group members. As before, already shared data in the past may still be available for other in the wild.

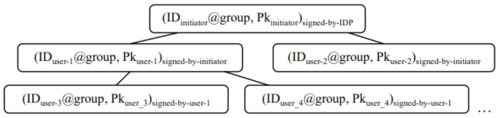
$(ID_{initiator}@group, Pk_{initiator})_{signed-by-IDP}$
$(ID_{user-1}@group, Pk_{user-1})_{signed-by-initiator}$
$(ID_{user-2}@group, Pk_{user-2})_{signed-by-initiator}$
$(ID_{user-3}@group, Pk_{user\_3})_{signed-by-user-1}$
$(ID_{user\_4}@group, Pk_{user\_4})_{signed-by-user-1}$
...

Fig. 9. An access graph representing the group with whom Alice shares certain data. Figure from the paper [31].

- **Security Analysis**
  We have seen that data remains encrypted at all times and Alice's as well as group member's private keys are never disclosed along this process such that passive attacks can be prevented as long as the identity provider prevents the creation of fake user key pairs.
  With re-encryption tokens Alice can control who has access to her data. These tokens simply re-encrypt the data to a format such that other users can decrypt it with their personal private keys. This protects her data at any time from the cloud service provider. These re-encryption tokens are not transitive and protects from malicious users to share her data with others using the same approach. This is not valid when Alice starts an in-situ re-keying for her new key pair.
  To sum up, Pilatus protects our data efficiently allowing encrypted data be shared among others in the cloud.

6. Conclusion

We have seen that privacy concerns are not negligible when reading the report [18] by M. Becker and reflecting over the actions taken by the European government so far. There exist several solutions how to protect the privacy of customers in the European Union. For this purpose, Pilatus provides a good ad-hoc solution , which is easily implementable by service providers. Nevertheless, Pilatus does not solve the dilemma of forced acceptance from the user's perspective. In the real world, software will evolve over time for which new updates will be issued. Moreover the companies' strategy may change and force users to accept newly released terms of use, if the customer wishes to continue using their services. The GDPR sets the limits. what companies may do with the data but it does not invert the power over the product, that companies hold so far.

The future will show how privacy will influence the market and the health wearable ecosystem over time as it does not really matter what was said along the investigation performed by M. Becker nor anywhere else if customers do not act accordingly. And companies do need the data to gain economic advantages over their competitors. Thus, even good solutions might not absolutely protect the customer's privacy if not applied and if no real monetary incentives exist. The GDPR provides incentives towards a sounder privacy protection by threatening with fines but the time will tell if fines provide the protection that does not hurt the European economy. In this era, data is crucial for the competitiveness of companies. And privacy simply does not provide this, yet.

References

[1] Apple Watch Series 3 - Technical Specifications. https://support.apple.com/kb/SP766?locale=en_US. 01.06.2018.
[2] Baby-step giant-step - Wikipedia. https://en.wikipedia.org/wiki/Baby-step_giant-step. Visited on 23.05.2018.
[3] Back up your Apple Watch. https://support.apple.com/en-us/ht204518. Visited on 13.06.2018.
[4] Best Android watches of 2018. https://www.androidauthority.com/best-android-watches-572773/. 29.04.2018.
[5] Blockchain - Wikipedia. https://en.wikipedia.org/wiki/Blockchain. Visited on 22.06.2018.
[6] data-protection-factsheet. https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf. 02.06.2018.
[7] Elliptic-curve cryptography. https://en.wikipedia.org/wiki/Elliptic-curve_cryptography. Visited on 14.06.2018.
[8] EUR-Lex - 32016R0679 - EN - EUR-Lex. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679. Visited on 13.06.2018.
[9] Fitbit Versa™ Smartwatch. https://www.fitbit.com/en-ca/shop/versa. 01.06.2018.
[10] Keybase. https://keybase.io/. Visited on 20.05.2018.
[11] Offizielle Website von Fitbit für Aktivitäts-Tracker und mehr. https://www.fitbit.com/de/home. 29.04.2018.
[12] Open Health Network. http://www.openhealth.cc/. Visited on 05.06.2018.
[13] Open Source Data Integration Tools — Open mHealht. http://www.openmhealth.org/. Visited on 19.06.2018.
[14] Quantified self. https://en.wikipedia.org/wiki/Quantified_self. 26.05.2018.
[15] Taraxa. http://taraxa.io/. 27.05.2018.
[16] Warum Fitbit. Visited on 13.06.2018.
[17] S. S. Adhatarao, M. Arumaithurai, D. Kutscher, and X. Fu. Isi: Integrate sensor networks to internet with icn. *IEEE Internet of Things Journal*, 5(2):491–499, 2018.
[18] M. Becker. Understanding users' health information privacy concerns for health wearables. In *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.
[19] F. M. Benčić and I. P. Žarko. Distributed ledger technology: Blockchain compared to directed acyclic graph. *arXiv preprint arXiv:1804.10013*, 2018.
[20] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on information theory*, 31(4):469–472, 1985.
[21] C. Gentry. *A fully homomorphic encryption scheme*. Stanford University, 2009.

[22] N. Karapanos, A. Filios, R. A. Popa, and S. Capkun. Verena: End-to-end integrity protection for web applications. In *Security and Privacy (SP), 2016 IEEE Symposium on*, pages 895–913. IEEE, 2016.

[23] G. Kenny and R. Connolly. Drivers of health information privacy concern: A comparison study. 2016.

[24] R. Ko and R. Choo. *The Cloud Security Ecosystem: Technical, Legal, Business and Management Issues*. Syngress, 2015. Chapter 5.

[25] J.-Y. Le Boudec, P. Thiran, and R. Urbanke. *Sciences de l'Information*. EPFL Press, 2014. Chapter 10.1.

[26] C. LeMahieu. Nano. url: https://nano.org. *Nano _whitepaper.pdf*, page 8, 2017.

[27] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 223–238. Springer, 1999.

[28] R. A. Popa, C. Redfield, N. Zeldovich, and H. Balakrishnan. Cryptdb: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, pages 85–100. ACM, 2011.

[29] S. Popov. The tangle. url: https://iota. org. *IOTA _ Whitepaper. pdf*, page 28, 2018.

[30] V. Rastogi and S. Nath. Differentially private aggregation of distributed time-series with transformation and encryption. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 735–746. ACM, 2010.

[31] H. Shafagh, A. Hithnawi, L. Burkhalter, P. Fischli, and S. Duquennoy. Secure sharing of partially homomorphic encrypted iot data. In *International Conference on Embedded Networked Sensor Systems (ACM SenSys 2017), November 5-8, 2017, Delft, The Netherlands*, 2017.

[32] H. Shafagh, A. Hithnawi, A. Dröscher, S. Duquennoy, and W. Hu. Talos: Encrypted query processing for the internet of things. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 197–210. ACM, 2015.

[33] S. Sharma and K. Chen. Privacy-preserving boosting with random linear classifiers for learning from user-generated data. *arXiv preprint arXiv:1802.08288*, 2018.

[34] Sherisse Pham. Facebook defends sharing user data with phone makers . http://money.cnn.com/2018/06/04/technology/facebook-apple-samsung-blackberry/index.html. Visited on 11.06.2018.

[35] E. Shi, H. Chan, E. Rieffel, R. Chow, and D. Song. Privacy-preserving aggregation of time-series data. In *Annual Network & Distributed System Security Symposium (NDSS)*. Internet Society., 2011.

[36] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini. A secure icn-iot architecture. In *Communications Workshops (ICC Workshops), 2017 IEEE International Conference on*, pages 259–264. IEEE, 2017.

[37] H. Zhang, Z. Wang, C. Scherb, C. Marxer, J. Burke, L. Zhang, and C. Tschudin. Sharing mhealth data via named data networking. In *Proceedings of the 3rd ACM Conference on Information-Centric Networking*, pages 142–147. ACM, 2016.

Appendix

A. Partially Homomorphic Encryption [24]

Homomorphic encryption allows us to manipulate encrypted data as it were unencrypted resulting in the same result when applying the same sequence of operations. Fully homomorphic encryption allows for a better utility but suffers from bad performance compared to partially homomorphic encryption. Therefore Pilatus and Talos use Partially Homomorphic Encryption (PHE) to encrypt the data with the consequence that a reduced set of operations can be applied on encrypted data, for example addition or multiplication. We will therefore explain Paillier and El Gamal Encryption as well as the difference between El Gamal and the Elliptic Curve (EC) version of the El Gamal cryptosystem. Both cryptosystems are based on asymmetric cryptography consisting of a key pair, one defined as private only known to the owner and its counterpart defined as public key, which is known to anybody. In order to access the public key of a receiver to whom we want to send a message, we need to ask for the public key. For this purpose an independent entity needs to exist who generates the key pair and distributes it accordingly and allows others to use the correct public key of the receiver.

Paillier cryptosystem

The Paillier cryptosystem leverages the fact, that computing the *n*th residue classes is computationally intensive and hence a hard problem. This cryptosystem allows users to perform homomorphic additions over encrypted data and provides the actual result once decrypted.

When generating the key pair for the Paillier cryptosystem, we need to take the following steps:

1) Select two large prime number p and q where the greatest common divisior of *p*q* and *(p-1)*(q-1)* is one
2) Calculate n = *p*q*
3) Calculate λ equals to the largest common multiple of p - *1* and q - *1*
4) Select g as a random integer where g $\in \mathbb{Z}^*{}_{n^2}$
5) Define L(x) = $\frac{x-1}{n}$
6) Check the existence of the modular multiplicative inverse such that n divides the order g by applying the following steps
7) u $= (L(g^\lambda \ mod \ n^2))^{-1} \ mod \ n$
8) The public key is (n, g)
9) The private key is (λ, u)

With the possession of the public and private key we are now able to perform the encryption and decryption. Now the health wearable user can encrypt the sensible data that can be securely stored onto the cloud by performing the following steps.

1) Select r as a random integer where r $\in \mathbb{Z}^*{}_{n^2}$
2) Calculate c = $g^m * r^n \ mod \ n^2$

with c being the encrypted data. As we can see, the message is encrypted as an exponent of g for which we need to keep in mind, that addition over encrypted data will be performed using multiplication as we will explain in the proof of the Paillier cryptosystem.
Whenever the health wearable user wishes to see its data, the user needs to download the appropriate data set and decrypt it, such that the data set is not in plain text in the cloud at any time making the encryption useless, otherwise. For the decryption only one step is performed with the result of getting the original message, which is possible by applying the private key. Without the private key, this is a computationally hard problem.

1) Calculate $m = L(c^{\lambda} mod n^2) * u mod n$

We now provide the proof that the Paillier cryptosystem actually works by showing that the key pair cancel each other out along the encryption and decryption process. Moreover we show, that the Paillier cryptosystem supports homomorphic addition over encrypted data as long as the result is not bigger than the modulus.
We will show that the original message equals the encrypted message once decrypted by simply applying the given algorithms and additional information which is stated in the original publication [27].

1) m = $L(c^{\lambda}$ mod $n^2)$ * u mod n
2) m = $L(c^{\lambda}$ mod $n^2)$ * $L(g^{\lambda}$ mod n^2 $)^{-1}$ mod n
3) m = $\frac{L(c^{\lambda} mod n^2)}{L(g^{\lambda} mod n^2)^{-1}}$ mod n
4) m = $\frac{\lambda[c]*1+n}{\lambda[g]*1+n}$ mod n (*by applying Lemma 10 in the original publication*)
5) m = $\frac{[c]*1+n}{[g]*1+n}$ mod n
6) m = $[c]_g$ mod n
7) m = m mod n (*as* $c = g^{[c]_g} * r^n$ *mod* n^2)

The encrypted message is encrypted as an exponent. In order to apply the addition over encrypted data, we need to multiply the encrypted messages with the same base. This allows the cryptosystem to add both messages as exponents, which will lead to the correct result. The base is g and as the values r_0 and r_1 are random as described in the encryption algorithm, both values can be combined in order to create another random value called r. We will show that the multiplication of encrypted data E(a) and E(b) is equal to the addition of a and b.

$$c = c_a * c_b \ mod \ n^2$$
$$c = g^a * {r_0}^n * g^b * {r_1}^n \ mod \ n^2$$
$$c = g^a * g^b * {r_0}^n * {r_1}^n \ mod \ n^2$$
$$c = g^{a+b} * r_0 * {r_1}^n \ mod \ n^2$$
$$c = g^{a+b} * r^n \ mod \ n^2$$

By applying the well defined decryption algorithm we get

$$D(c) = a + b$$

and we have shown that the Paillier cryptosystem supports homomorphic encryption over encrypted data.

El Gamal cryptosystem

This cryptosystem is based on the hardness of solving discrete logarithms. El Gamal supports homomorphic multiplication operations on encrypted data. As in the Paillier cryptosystem, we need to generate the key pair by applying the following steps:

1) Select a large prime *p*
2) Select a primitive value α in modulo *p*
3) Randomly select *d* so that 2 $\leq$ *d* $\leq$ *p* - 2
4) Calculate β = α^d mod *p*
5) The public key is (*p*, α, β)
6) The private key is *d*

As in the Paillier cryptosystem the public and private key allows us to perform the encryption and decryption. In order to encrypt the data M being smaller than *p*, we perform the following steps:

1) Select a random integer *k* which remains private at all times
2) Calculate r = α^k mod *p*
3) Calculate t = β^k * M mod p
4) Discard k which should not be published in any way
5) And voilà, the encrypted message equals (*r, t*)

If we now leak k along these steps, it makes it easy for an attacker to decrypt (*r, t*) into M without the private key as the strength of this cryptosystem is based upon the fact that solving for k is a hard problem.
The decryption of (*r, t*) into M is performed by applying the private key *d* and solving the following formula.

1) Calculate M = *t* * r^{-d} mod p

of which *r* and *p* are known and *d* is only known to the user, if the data was encrypted using its known public key.

The proof of the El Gamal cryptosystem allows us to see that the public and private key cancel each other out when decrypting the message. The main strength of this algorithm is, that it is a hard problem to figure out k which is never leaked along the encryption and decryption as stated before.

1) M = *t* * r^{-d} mod *p*
2) M = β^k * M * $(\alpha^k)^{-d}$ mod *p*
3) M = $(\alpha^d)^k$ * M * $(\alpha^k)^{-d}$ mod *p*
4) M = M * $(\alpha^{dk}$ * $\alpha^{-dk})$ mod *p*
5) M = M * 1 mod *p*

El Gamal supports homomorphic multiplication on encrypted data as long as the result is not bigger than the modulus p. We will show that the multiplication of encrypted data equals the result of the multiplication of the same data unencrypted.

$$D(E(a) \times E(b)) \; mod \; p = t_a \times t_b \times (r_a \times r_b)^{-d} \; mod \; p \tag{1}$$

We now provide the proof.

$$\begin{aligned} \rightarrow r &= r_a * r_b \; mod \; p \\ &= \alpha^{k_a} * \alpha^{k_b} \; mod \; p \\ &= \alpha^{k_a + k_b} \; mod \; p \\ &= \alpha^k \end{aligned}$$

Furthermore we can modify transform the t s in the same manner such that

$$\begin{aligned} t &= t_a * t_b \; mod \; p \\ &= (\beta^{k_a} * M_a) * (\beta^{k_b} * M_b) \; mod \; p \\ &= \beta^{k_a + k_b} * M_a * M_b \; mod \; p \\ &= \beta^k * M_a * M_b \; mod \; p \end{aligned}$$

As we have shown how to transform the r and t values we can subsitute them in the decryption formula, we get

$$\begin{aligned} D(c) &= (\beta^k * M_a * M_b) * (\alpha^k)^{-d} \; mod \; p \\ &= (\alpha^d)^k * M_a * M_b * (\alpha^k)^{-d} \; mod \; p \\ &= M_a * M_b * (\alpha^{d*k} * \alpha^{-d*k}) \; mod \; p \\ &= a * b \; mod \; p \end{aligned}$$

Which concludes the proof.

In Pilatus, the creators apply Elliptic Curve (EC) El Gamal in order to establish additive homomorphic properties and therefore not homomorphic multiplication on encrypted data. EC El Gamal is based on the algebraic structure of elliptic curves over finite fields. Therefore we can work with smaller numbers which allows this modified cryptosystem to work with smaller keys compared to El Gamal and making it more suitable for the IoT [7]. As an example, we can use for a 4096 Bit cryptosystem a key of below 400 Bit in EC cryptosystems.
In EC El Gamal the following steps are performed, whenever a sender S wants to encrypt a message for its receiver R. First the receiver R must provide some parameters, with which the sender S can encrypt and send the message M for R.

1) Receiver chooses E mod p with E being the elliptic curve and p a large prime number,
2) then chooses a point α (a value) on the elliptic curve E.
3) Furthermore R chooses a secret value a known only to itself and
4) computes β = α * a

The parameters α and β are made public and consequently can be retrieved by the sender S in order to encrypt the message M.

1) Sender chooses a random integer $k \in \mathbb{Z}$,
2) computes y_1 = k*α and y_α = M + k*β
3) and finally sends (y_1, y_α) to receiver R.

Now the receiver R gets the encrypted message as (y_1, y_α) and performs the following calculation

$$\begin{gathered} y_\alpha - a * y_1 = \\ M + k * \beta - a * k * \alpha = \\ M + k * \alpha * a - a * k * \alpha = M \end{gathered}$$

and is able to decrypt the original message M. The proof for homomorphic addition on encrypted data in the EC El Gamal cryptosystem is omitted.

B. Chinese Remainder Theorem [25]

The Chinese Remainder Theorem allows Pilatus to optimize the en- and decryption in the EC El Gamal cryptosystem which is especially used for solving congruences. Consider the following function with m_1, m_2 $\in \mathbb{N}$ and ϕ a function defined as

$$\begin{aligned} &\phi : \mathbb{Z}_{0 \; \leq \; k \; < \; m_1 m_2} \rightarrow \mathbb{Z}_{0 \; \leq \; i \; < \; m_1} \times \mathbb{Z}_{0 \; \leq \; j \; < \; m_2} \\ &\phi : [k]_{m_1 m_2} \longmapsto ([k]_{m_1}, [k]_{m_2}) \end{aligned}$$

where the Chinese Remainder Theorem checks if ϕ is actually a bijective function.

Chinese Remainder Theorem: Let m$_1$ and m_2 be two integers both being $\geq$ 2.

1) If the greatest common divisor of m_1 and m_2 is 1, then ϕ is defined to be bijective and it is furthermore isomorphic for addition and multiplication.
2) If the greatest common divisor of m_1 and m_2 is not 1, then ϕ is neither a surjective nor injective function.

Thanks to this theorem Pilatus can apply an optimized version of EC El Gamal cryptosystem for IoT by splitting the data into smaller chunks and performing the encryption and decryption without scrambling with the topology when disassembling and assembling these data chunks and consequently ensure data integrity.