A security survey of middleware for the Internet of Things

Paul Fremantle, Philip Scott

The rapid growth of small Internet connected devices, known as the Internet of Things (IoT), is creating a new set of challenges to create secure, private infrastructures. The purpose of this paper is to review the current literature on the challenges and approaches to security and privacy in the Internet of Things, with an especial focus on how these aspects are handled in IoT middleware. We focus on IoT middleware because many systems are built from existing middleware and these inherit the underlying security properties of the middleware framework. The paper is composed of three main sections. Firstly, we look at the general security and privacy challenges around IoT. Secondly, we present a structured literature review of the available middleware and how security is handled in these middleware approaches. Finally, we draw a set of conclusions and identify further work in this area.

A security survey of middleware for the Internet of Things

Paul Fremantle[1] **and Philip Scott**[1]

[1]**University of Portsmouth, Portsmouth, UK**

ABSTRACT

The rapid growth of small Internet connected devices, known as the Internet of Things (IoT), is creating a new set of challenges to create secure, private infrastructures. The purpose of this paper is to review the current literature on the challenges and approaches to security and privacy in the Internet of Things, with an especial focus on how these aspects are handled in IoT middleware. We focus on IoT middleware because many systems are built from existing middleware and these inherit the underlying security properties of the middleware framework.

The paper is composed of three main sections. Firstly, we look at the general security and privacy challenges around IoT. Secondly, we present a structured literature review of the available middleware and how security is handled in these middleware approaches. Finally, we draw a set of conclusions and identify further work in this area.

Keywords: Security, Privacy, Internet of Things, IoT, Middleware

INTRODUCTION

The *Internet of Things (IoT)* was originally coined as a phrase by Kevin Ashton in 1990 [7], with reference to "taggable" items that used RFID chips to become electronically identifiable and therefore amenable to interactions with the Internet. With the ubiquity of cheap processors and System-on-Chip (SoC) based devices, the definition has expanded to include wireless and internet-attached sensors and actuators, including smart meters, home automation systems, internet-attached set-top-boxes, smartphones, connected cars, and other systems that connect the physical world to the Internet either by measuring it or affecting it.

The Internet of Things is defined variously, but we will use the definition from [31]:

> "A dynamic global network infrastructure with self configuring capabilities based on standard and interoperable communication protocols where physical and virtual 'things' have identities, physical attributes, and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network."

The number of IoT devices has grown rapidly, with a recent estimate suggesting that there were 12.5 billion internet attached devices in 2010 and a prediction of 50 billion devices by 2020 [43]. This brings with it multiple security challenges:

- These devices are becoming more central to people's lives, and hence the security is becoming more important.

- These devices, due to size and power limitations, may not support the same level of security that we would expect from more traditional Internet-connected devices.

- The sheer scale and number of predicted devices will create new challenges and require new approaches to security.

We will explore each of these areas in more detail below, as we look at the potential attacks that are possible on IoT devices.

Security Characteristic	Device / Hardware	Network	Cloud / Server-Side
Confidentiality	A. Hardware attacks	B. Encryption with low capability devices	C. Privacy concerns
Integrity	D. Lack of attestation, illicit updates	E. Signatures with low capability devices	F. Unchanged
Availability	G. Physical attacks; Radio jamming	H. Unreliable networks	I. Unchanged
Authentication	J. Lack of user input; Hardware retrieval of keys	K. Challenges of using federated identity	L. Lack of widely implemented standards around Device Identity
Access Control	M. Physical access; Lack of local authentication	N. Lightweight protocols for access control	O. Requirement for user managed access controls
Non-Repudiation	P. No secure local storage; Low capability devices	Q. Signatures with low capability devices	R. Unchanged

Figure 1. Matrix of security challenges for the IoT

1 APPROACH

In order to understand the security threats against the Internet of Things, we need to take an approach to classifying threats. The most widely used ontology of security threats is based on the classic "CIA" model (Confidentiality, Integrity, Availability) [96] which has been extended over the years and is now often referred to as the "CIA+" model [111]. In the course of reviewing the available literature and approaches to IoT security, we have created a proposed expansion of the existing framework that we believe works better in the IoT space. In particular, we propose a new ontology based on a matrix of evaluation where we look at each of the classic security challenges in three different aspects: device/hardware, network, and cloud/server-side. In some cells in this matrix, we have not identified any areas where the IoT space presents new challenges: in other words, whilst the domain space covered by these cells contains security challenges, those challenges are no different from existing Web and Internet security challenges in that domain. In those cells we can say that the challenges are "unchanged". In other cells we can identify the further challenges posed by the challenges unique to (or extended by) by the Internet of Things.

Figure 1 shows the matrix we will use for evaluating security challenges. In each cell we summarise the main challenges that are *different* in the IoT world or at least exacerbated by the challenges of IoT compared to existing Internet security challenges. We will explore each cell in the matrix in detail below. Each of the cells is given a letter from *A* to *R* and these letters are used as a key to refer to the cells below.

The three aspects (Hardware/Device, Network, Cloud/Server) were chosen because as we read the available literature these areas became clear as a way of segmenting the unique challenges within the context of the IoT compared with existing Internet security challenges. These form a clear logical grouping of the different assets involved in IoT systems. We will provide a quick overview of each area before we look in detail at each cell of the matrix.

Device and Hardware

IoT devices have specific challenges that go beyond those of existing Internet clients. These challenges come from: the different form factors of IoT devices; from the power requirements of

IoT devices; and from the hardware aspects of IoT devices. The rise of cheap mobile telephony has driven down the costs of 32-bit processors (especially those based around the ARM architecture [48]), and this is increasingly creating lower cost microcontrollers and System-on-Chip (SoC) devices based on ARM. However, there are still many IoT devices built on 8-bit processors, and occasionally, 16-bit [118]. In particular the open source hardware platform Arduino [6] supports both 8-bit and 32-bit controllers, but the 8-bit controllers remain considerably cheaper and more popular.

Network

IoT devices may use much lower power, lower bandwidth networks than existing Internet systems. Cellular networks often have much higher latency and more "dropouts" than fixed networks [27]. The protocols that are used for the Web are often too data-intensive and power-hungry for IoT devices. Network security approaches such as encryption and digital signatures are difficult or impractical in small devices.

Cloud/Server-Side

While many of the existing challenges apply here, there are some aspects that are exacerbated by the IoT for the server-side or cloud infrastructure. These include: the often highly personal nature of data that is being collected and the requirement to manage privacy; the need to provide user-managed controls for access; and the lack of clear identities for devices making it easier to spoof or impersonate devices.

1.1 Cell A: Confidentiality / Hardware

The confidentiality of data on the device itself can certainly be an issue. Whilst it could be argued that many IoT devices are in public areas, even these devices may store historical data locally, or may have security data (e.g. keys, passwords, credentials or sensitive code) that is liable to attack. [112] is a comprehensive study of many semi-invasive attacks that can be done on hardware. [121, 76] cover the concept of *side channel attacks* where the power usage or other indirect information from the device can be used to steal information.

A related issue to confidentiality of the data on the device is the challenges inherent in updating devices and pushing keys out to devices.

The distribution and maintenance of certificates and public-keys onto embedded devices is complex [119]. In addition, sensor networks may be connected intermittently to the network resulting in limited or no access to the Certificate Authority (CA). To address this, the use of threshold cryptographic systems that do not depend on a single central CA has been proposed [122], but this technology is not widely adopted: in any given environment this would require many heterogeneous Things to support the same threshold cryptographic approach.

We can also see clearly from a number of publicised attacks [79, 64, 58] that device designers have not adjusted to the challenges of designing devices that will be connected either directly or indirectly to the internet.

A further security challenge for confidentiality and hardware is the fingerprinting of sensors or data from sensors. In [20] it has been shown that microphones, accelerometers and other sensors within devices have unique "fingerprints" that can uniquely identify devices.

Finally, the use of PKI requires devices to be updated as certificates expire. The complexity of performing updates on IoT devices is harder, especially in smaller devices where there is no user interface. For example, many devices need to be connected to a laptop in order to perform updates. This requires human intervention and validation, and in many cases this is another area where security falls down. For example, many situations exist where security flaws have been fixed but because devices are in homes, or remote locations, or seen as appliances rather than computing devices, updates are not installed [58].

1.2 Cell B: Confidentiality / Network

The confidentiality of data on the network is usually protected by encryption of the data. There are a number of challenges with using encryption in small devices. Performing public key encryption on 8-bit microcontrollers has been enhanced by the use of Elliptical Curve Cryptography (ECC) [65, 81]. ECC reduces the time and power requirements for the same level of encryption as an equivalent RSA public-key encryption [100] by an order of magnitude [53, 108, 109]: RSA encryption on constrained

8-bit microcontrollers may take minutes to complete, whereas similar ECC-based cryptography completes in seconds. However, despite the fact that ECC enables 8-bit microcontrollers to participate in public-key encryption systems, in many cases it is not used. We can speculate as to why this is: firstly, as evidenced by [109], the encryption algorithms consume a large proportion of the available ROM on small controllers. Secondly, there is a lack of standard open source software. For example, a search that we carried out (on the 21st April 2015) of the popular open source site Github for the words "Arduino" and "Encryption" revealed 10 repositories compared to "Arduino" and "HTTP" which revealed 467 repositories. However, recently an open source library for AES on Arduino [68] has made the it more effective to use cryptography on Atmel-based hardware.

Another key challenge in confidentiality is the complexity of the most commonly used encryption protocols. The standard Transport Layer Security (TLS [37]) protocol can be configured to use ECC, but even in this case the handshake process requires a number of message flows and is sub-optimal for small devices as documented in [66]. [93] has argued that using TLS with *Pre-Shared Keys* (PSK) improves the handshake. However, they fail to discuss in any detail the significant challenges with using PSK with IoT devices: the fact that either individual symmetric keys need to be deployed onto each device during the device manufacturing process, or the same key re-used. In this case there is a serious security risk that a single device will be broken and thus the key will be available.

There is an alternative protocol for UDP: DTLS (Datagram Transport Level Security) [99] which provides a lighter weight approach than TLS. However, there is still a reasonably large RAM and ROM size required for this [63], and this requires that messages be sent over UDP which has significant issues with firewalls and home routers, making it a less effective protocol for IoT applications [11]. There is ongoing work at the IETF to produce an effective profile of both TLS and DTLS for the IoT [115].

A significant area of challenge for network confidentiality in IoT is the emergence of new radio protocols for networking. Previously there were equivalent challenges with Wifi networks as protocols such as WEP were broken citepcam2003security, and there are new attacks on protocols such as Bluetooth 4.0 (also known as Bluetooth LE/BLE). For example, while BLE utilises AES encryption which has a known security profile, a new key exchange protocol was created, which turns out to be flawed, allowing any attacker present during key exchange to intercept all future communications [101]. One significant challenge for IoT is the length of time it takes for vulnerabilities to be addressed when hardware assets are involved. While the BLE key exchange issues are addressed in the latest revision of BLE, we can expect it to take a very long time for the devices that encode the flawed version in hardware to be replaced. By analogy, many years after the WEP issues were uncovered, in 2011 a study showed that 25% of wifi networks were still at risk [21].

In [97] a theoretical model of traceability of IoT devices and particularly RFID systems is proposed in order to prevent unauthorised data being accessible. A protocol that preserves the concept of untraceability is proposed.

Many of the same references and issues apply to section *E* where we look at the use of digital signatures with low power devices.

1.3 Cell C: Confidentiality & Cloud/Server

While the issues here are largely similar to normal web- and internet-based systems, there are certainly concerns about individuals privacy on with the Internet of Things. For example, the company Fitbit [44] made data about users sexual activity available and easily searchable online [123] by default. There are social and policy issues regarding the ownership of data created by IoT devices [98, 86]. We address these issues in more detail in the cell *O* where we look at the access control of IoT data and systems in the cloud and on the server-side.

A second concern that is exacerbated by the Internet of Things are concerns with correlation of data and metadata, especially around *de-anonymisation*. In [87] it was shown that anonymous metadata could be de-anonymized by correlating it with other publicly available social metadata. This is a significant concern with IoT data. This is also closely related to the fingerprinting of sensors within devices as discussed in cell *A*. An important model for addressing these issues in the cloud are systems that filter, summarise and use *stream-processing* technologies to the data coming from IoT devices before this data is more widely published. For example, if we only publish a summarised co-ordinate rather than the raw accelerometer data we can potentially avoid fingerprinting de-anonymisation attacks.

In addition, an important concern has been raised in the recent past with the details of the government

sponsored attacks from the US National Security Agency (NSA) and British Government Communications Headquarters (GCHQ) that have been revealed by Edward Snowden [25]. These bring up three specific concerns on IoT privacy and confidentiality.

The first concern is the revelations that many of the encryption and security systems have had deliberate backdoor attacks added to them so as to make them less secure [69]. The second concern is the revelation that many providers of cloud hosting systems have been forced to hand over encryption keys to the security services [71]. The third major concern is the revelations on the extent to which metadata is utilised by the security services to build up a detailed picture of individual users [13].

The implications of these three concerns when considered in the light of the Internet of Things is clear: a significantly deeper and larger amount of data and metadata will be available to security services and to other attackers who can utilize the same weaknesses that the security services compromise.

1.4 Cell D: Integrity & Hardware/Device

The concept of integrity refers to maintaining the accuracy and consistency of data. In this cell of the matrix, the challenges are in maintaining the device's code and stored data so that it can be trusted over the lifecycle of that device. In particular the integrity of the code is vital if we are to trust the data that comes from the device or the data that is sent to the device. The challenges here are viruses, firmware attacks and specific manipulation of hardware. For example, [51] describes a worm attack on router and IoT firmware.

The traditional solution to such problems is attestation [102, 23, 107]. Attestation is important in two ways. Firstly, attestation can be used by a remote system to ensure that the firmware is unmodified and therefore the data coming from the device is accurate. Secondly, attestation is used in conjunction with hardware-based secure storage (Hardware Security Managers, as described in [35] to ensure that authentication keys are not misused. The model is as follows.

In order to preserve the security of authentication keys in a machine where human interaction is involved, the user is required to authenticate. Often the keys are themselves encrypted using the human's password or a derivative of the identification parameters. However, in an unattended system, there is no human interaction. Therefore the authentication keys need to be protected in some other way. Encryption on its own is no help, because the encryption key is then needed and this becomes a circular problem. The solution to this is to store the authentication key in a dedicated hardware storage. However, if the firmware of the device is modified, then the modified firmware can read the authentication key, and offer it to a hacker or misuse it directly. The solution to this is for an attestation process to validate the firmware is unmodified before allowing the keys to be used. Then the keys must also be encrypted before sending them over any network.

These attestation models are promoted by groups such the Trusted Computing Group [113], and Samsung Knox [105]. These rely on specialized hardware chips such as the Atmel AT97SC3204 [9] which implement the concept of a Trusted Platform Module [83]. There is research into running these for Smart Grid devices [92]. However, whilst there is considerable discussion of using these techniques with IoT, during our literature review we could not find evidence of any real-world devices apart from those based on mobile-phone platforms (e.g. phones and tablets) that implemented trusted computing and attestation.

1.5 Cell E: Integrity & Network

Maintaining integrity over a network is managed as part of the public-key encryption models by the use of digital signatures. The challenges for IoT are exactly those we already identified in the section B above where we described the challenges of using encryption from low-power IoT devices.

1.6 Cell F: Integrity & Cloud/Server

This area is unchanged by the IoT.

1.7 Cell G: Availability & Device/Hardware

One of the significant models used by attackers is to challenge the availability of a system, usually through a Denial of Service (DoS) or Distributed Denial of Service (DDoS) attack. DoS attacks and availability attacks are used in several ways by attackers. Firstly, there may be some pure malicious or destructive urge (e.g. revenge, commercial harm, share price manipulation) in bringing down a system. Secondly, availability attacks are often used as a pre-cursor to an authentication or spoofing attack.

IoT devices have some different attack vectors for availability attacks. These include resource consumption attacks (overloading restricted devices), physical attacks on devices.

1.8 Cell H: Availability & Network

There are clearly many aspects of this that are the same as existing network challenges. However, there are some issues that particularly affect IoT. In particular, there are a number of attacks on local radio networks that are possible. Many IoT devices use radio networking (Bluetooth, Wifi, 3G, GPRS, LoRa and others) and these can be susceptible to radio jamming. Another clear area of attack is simply physical access. For example, even wired networks are much more susceptible to DoS attacks when the devices are spread widely over large areas.

1.9 Cell I: Availability & Cloud/Server

This area is unchanged by the IoT.

1.10 Cell J: Authentication & Device/Hardware

We will consider the authentication of the device to the rest of the world in later sections. In this cell of the matrix we must consider the challenges of how users or other devices can securely authenticate to the device itself. These are however related: a user may bypass or fake the authentication to the device and thereby cause the device to incorrectly identify itself over the network to other parts of the Internet.

Some attacks are very simple: many devices come with default passwords which are never changed by owners. In a well-publicised example [58], a security researcher gained access to full controls of a number of "smart homes".

Similarly many home routers are at risk through insecure authentication [5]. Such vulnerabilities can then spread to other devices on the same network as attackers take control of the local area network.

A key issue here is the initial *registration* of the device. A major issue with hardware is when the same credential, key, or password is stored on many devices. Devices are susceptible to hardware attacks (as discussed above) and the result is that the loss of a single device may compromise many or all devices. In order to prevent this, devices must either be pre-programmed with unique identifiers and credentials at manufacturing time, or must go through a registration process at setup time. In both cases this adds complexity and expense, and may compromise usability. We argued for the use of the Dynamic Client Registration process in [46] to create unique keys/credentials for each device.

1.11 Cell K: Authentication & Network

Unlike browsers or laptops where a human has the opportunity to provide authentication information such as a userid and password, IoT devices normally run unattended and need to be able to power-cycle and reboot without human interaction. This means that any identifier for the device needs to be stored in the program memory (usually SRAM), ROM or storage of the device. This brings two distinct challenges:

- The device may validly authenticate, but its code may have been changed.

- Another device may steal the authentication identifier and may *spoof* the device.

In the Sybil attack [89] a single node or nodes may impersonate a large number of different nodes thereby taking over a whole network of sensors.

In all cases, attestation is a key defence against these attacks.

Another defence is the use of *reputation* and reputational models to associate a trust value to devices on the network.

Reputation is a general concept widely used in all aspects of knowledge ranging from humanities, arts and social sciences to digital sciences. In computing systems, reputation is considered as a *measure* of how trustworthy a system is. There are two approaches to trust in computer networks: the first involves a "black and white" approach based on security certificates, policies, etc. For example, SPINS [95], develops a trusted network. The second approach is probabilistic in nature, where trust is based on reputation, which is defined as a probability that an agent is trustworthy. In fact, reputation is often seen as one measure by which trust or distrust can be built based on good or bad past experiences and observations (direct trust) [62] or based on collected referral information (indirect trust) [1].

In recent years, the concept of reputation has shown itself to be useful in many areas of research in computer science, particularly in the context of distributed and collaborative systems, where interesting

issues of trust and security manifest themselves. Therefore, one encounters several definitions, models and systems of reputation in distributed computing research (e.g. [47, 62, 110]).

There is considerable work into reputation and trust for wireless sensor networks, much of which is directly relevant to IoT trust and reputation. The Hermes and E-Hermes [125, 126] systems utilise Bayesian statistical methods to calculate reputation based on how effectively nodes in a mesh network propogate messages including the reputation messages. Similarly, [29] evaluates reputation based on the packet-forwarding trustworthiness of nodes, in this case using fuzzy logic to provide the evaluation framework. Another similar work is [80] which again looks at the packet forwarding reputation of nodes.

1.12 Cell L: Authentication & Cloud/Server

The IETF has published a draft guidance on security considerations for IoT [90]. This draft does discuss both the bootstrapping of identity and the issues of privacy-aware identification. One key aspect is that of bootstrapping a secure conversation between the IoT device and other systems, which includes the challenge of setting-up an encrypted and/or authenticated channel such as those using TLS, HIP or Diet HIP. The Host Identity Protocol (HIP) [85] is a protocol designed to provide a cryptographically secured endpoint to replace the use of IP addresses, which solves a significant problem – IP-address spoofing – in the Internet. Diet HIP [84] is a lighter-weight rendition of the same model designed specifically for IoT and M2M interactions. While HIP and Diet HIP solve difficult problems, they have significant disadvantages to adoption. Firstly, they require low-level changes within the IP stack to implement. Secondly, as they replace traditional IP addressing they require a major change in many existing systems to work. In addition, neither HIP nor Diet HIP address the issues of federated authorization and delegation.

We proposed [45] using *federated* identity protocols such as OAuth2 [54] with IoT devices, especially around the MQTT protocol [75]. The IOT-OAS [30] work similarly addresses the use of OAuth2 with CoAP. Other related works include the work of Augusto et al. [12] have built a secure mobile digital wallet by using OAuth together with the XMPP protocol [103]. In [46], we extended the usage of OAuth2 for IoT devices to include the use of Dynamic Client Registration [104] which allows each device to have its own unique identity, which we discussed as an important point in the section about Cell A.

1.13 Cell M: Access Control & Device/Hardware

There are two challenges to access control at the device level. Firstly, devices are often physically distributed and so an attacker is likely to be able to gain physical access to the device. The challenges here were already discussed in Cell A. However, there is a further challenge: access control requires a concept of identity. We cannot restrict or allow access without some form of authentication to the device, and as discussed in our review of Cell J, this is a significant challenge. In addition, systems such as Webinos [36] have proposed using policy-based access control mechanisms such as XACML [50] for IoT devices. However, XACML is relatively heavyweight and expensive to implement [116], especially in the context of low power devices. To address this, Webinos has developed an engine which can calculate the subset of the policy that is relevant to a particular device. Despite this innovation, the storage, transmission and processing costs of XACML are very high for an IoT device.

1.14 Cell N: Access Control & Network

There are a number of researchers looking at how to create new lightweight protocols for access control in IoT scenarios. [78] describe a new protocol for IoT authentication and access control is proposed based on ECC with a lightweight handshake mechanism to provide an effective approach for IoT, especially in mobility cases. [57] propose a non-centralised approach for access control that uses ECC once again and supports capability tokens in the CoAP protocol.

1.15 Cell O: Access Control & Cloud/Server

The biggest challenge for privacy is ensuring access control at the server or cloud environment of data collected from the IoT. There is some significant overlap with the area of confidentiality of data in the cloud as well (Cell *C*).

We argued strongly in [45] that existing hierarchical models of access control are not appropriate for the scale and scope of the IoT. There are two main approaches to address this. The first is *policy-based* security models where roles and groups are replaces by more generic policies that capture real-world requirements such as "A doctor may view a patient's record if they are treating that patient in the emergency room". The second approach to support the scale of IoT is user-directed security controls. In [41] a strong

case is made for ensuring that users can control access to their own resources and to the data produced by the IoT that relates to those users. The User Managed Access (UMA) from the Kantara Initiative enhances the OAuth specification to provide a rich environment for users to select their own data sharing preferences [60]. We would argue strongly that this overall concept of user-directed access control to IoT data is one of the most important approaches to ensuring privacy.

[120] argues that contextual approaches must be taken to ensure privacy with the IoT. Many modern security systems use context and reputation to establish trust and to prevent data leaks. Context-based security [82] defines this approach which is now implemented by major Web systems including Google and Facebook.

1.16 Cell P: Non-Repudiation & Device/Hardware

The biggest challenge in the non-repudiation network with IoT devices is the challenge of using *attestation* for small devices. Attestation is discussed in detail in Cell *D*. Without attestation, we cannot trust that the device system has not been modified and therefore it is not possible to trust any non-repudiation data from the device.

1.17 Cell Q: Non-Repudiation & Network

The same challenges apply here as discussed in cells *B, E*. Non-repudiation on the wire requires cryptography techniques and these are often hindered by resource restrictions on small devices. In [91] a non-repudiation protocol for restricted devices is proposed.

1.18 Cell R: Non-Repudiation & Cloud/Server

This area is unchanged by the IoT.

1.19 Summary of the review of security issues

In this section we have proposed a widened ontology for evaluating the security issues surrounding the Internet of Things, and examined the existing literature and research in each of the cells of the expanded matrix. This is an important basis for the next section where we examine the provisions around security and privacy that are available in available middleware for the Internet of Things.

One area that crosses most or all of the cells in our matrix is the need for a holistic and studied approach to enabling privacy in the IoT. As discussed in a number of cells, there are significant challenges to privacy with the increased data and metadata that is being made available by IoT-connected devices. An approach that has been proposed to address this is *Privacy by Design* [26]. This model suggests that systems should be designed from the ground up with the concept of privacy built into the heart of each system. Many systems have added security or privacy controls as "add-ons", with the result that unforeseen attacks can occur.

In reviewing these areas, we identified a list of security properties and capabilities that are important for the security and privacy of IoT. We will use this list in the second part of this paper as columns in a new table where we evaluate a set of middleware on their provision of these capabilities.

Integrity and Confidentiality
> The requirement to provide integrity and confidentiality is an important aspect in any network and as discussed in cells *A-E* there are a number of challenges in this space for IoT.

Access Control
> Maintaining access control to data that is personal or can be used to extract personal data is a key aspect of privacy. In addition, it is of prime importance with actuators that we do not allow unauthorised access to control aspects of our world.

Policy-based security
> Managing security in the scale of IoT is unfeasible in a centralised approach. As we discussed, access control and identity models need to be based on policies such as XACML rather than built in a traditional hierarchical approach.

Authentication
> Clearly, in order to respect privacy, IoT systems need a concept of authentication.

Federated Identity

    As argued in cell *L*, there is a clear motivation for the use of federated models of identity for authentication in IoT networks.

Attestation

    Attestation is an important technique to prevent tampering and hence issues with integrity of data as well as confidiality in IoT.

Summarisation and Filtering

    The need to prevent de-anonymisation is a clear driver for systems to provide summarisation and filtering technologies such as stream processing.

Privacy by Design

    As discussed above, an important approach to ensuring privacy is to build this into the design of the systems.

Context-based security and Reputation

    Many modern security models adapt the security based on a number of factors, including location, time of day, previous history of systems, and other aspects known as context. Another related model is that of the reputation of systems, whereby systems that have unusual or less-than-ideal behaviour can be trusted less using probabilistic models. In both cases there are clear application to IoT privacy as discussed above.

There are of course many other aspects to IoT security and privacy as we have demonstrated in the matrix table and accompanying description of each cell. However, these specific aspects for an effective set of criteria by which to analyse different systems, as we show below.

2 SECURE MIDDLEWARE FOR THE INTERNET OF THINGS

2.1 Introduction

Middleware has been defined as computer software that has an intermediary function between the various applications of a computer and its operating system [55]. In our case, we are interested in middleware that is specifically designed or adapted to provide capabilities for IoT networks. There are a number of existing surveys of IoT middleware.

Bandyopadhyay et al. [15, 14] review a number of middleware systems designed for IoT systems. While they look at security in passing, there is no detailed analysis of the security of each middleware system. [28] calls out the need for security, but no analysis of the approaches or existing capabilities is provided. [10] is a very broad survey paper that addresses IoT middleware loosely.

It is clear then, that a detailed evaluation of security in IoT middleware is a useful contribution to the literature. We therefore identified a set of middleware systems to study.

2.2 Review Methodology

This set was identified through a combination of the existing literature reviews on IoT middleware [15, 28] together with our own search for middleware systems that explicitly target IoT scenarios. Some of the systems that were included in these papers we excluded from our list on the basis that they were not middleware. For example, [28] lists TinyREST [77] as a middleware, but in fact we considered this paper to be the definition of a standard protocol and therefore we excluded it.

Our search strategy was to use a search for the terms ("IoT" OR "Internet of Things") AND "Middleware". We searched only in the subject terms and restricted the search to academic papers written in English. The search was carried out by the Portsmouth University Discovery system which is a metasearch engine. The list of databases that are searched is available at [117]. This strategy identified 152 papers. We then manually reviewed the abstracts of these papers to identify a list of functioning middleware systems as opposed to papers that describe other aspects of IoT without describing a middleware system. This produced a list of 22 middleware systems. The table in Figure 2 summarises the middleware systems that we reviewed as well as the major findings of the review.

In our study, we looked for the security properties listed in section 1.19. We also identified if the middleware had a clearly defined security model and/or security implementation. In addition, we used

some other more general characteristics, including whether the systems supported SOAP/Web Services, REST, Event-Based models, Semantic approaches, and IoT-specific protocols. Together with the security properties these make up the columns of our summary table.

Middleware	Defined Security Model	Tangible Security Architecture	SOAP/WS-*	REST	Event Driven	Semantic	IoT-specific Protocol Support	Integrity and Confidentiality	Access Control	User-centric access control	Policy-based security	Authentication	Federated Identity	Attestation	Summarisation and Filtering	Privacy By Design	Context-based security/Reputation
ASPIRE	N	N	Y	N	N	N	N	-	-	-	-	-	-	-	-	-	-
CBCPM	N	N	N	Y	Y	N	N	-	-	-	-	-	-	-	-	-	-
Dioptase	N	N	N	N	Y	Y	N	-	-	-	-	-	-	-	Y	-	-
DREMS	Y	Y	N	N	Y	N	Y	Y	Y	N	N	Y	N	N	N	N	N
EDSOA	N	N	Y	N	Y	N	N	-	-	-	-	-	-	-	-	-	-
GSN	N	N	N	N	Y	N	N	-	-	-	-	-	-	-	Y	-	-
Hydra/Linksmart	Y	Y	Y	N	N	Y	N	Y	Y	N	N	Y	N	N	N	N	N
ISMB/VIRTUS	Y	Y	N	N	Y	N	N	Y	Y	N	N	Y	Y	N	N	N	N
MOSDEN	N	N	N	N	Y	N	N	-	-	-	-	-	-	-	Y	-	-
NAPS	Y	N	-	-	Y	N	Y	-	-	-	-	-	-	-	-	-	-
OpenIoT	N	N	-	-	-	Y	N	-	-	-	-	-	-	-	Y	-	-
SBIOTCM	N	N	Y	N	N	N	N	-	-	-	-	-	-	-	-	-	-
SIRENA	Y	Y	Y	N	N	N	N	Y	N	N	N	Y	N	N	N	N	N
SMEPP	Y	Y	-	-	Y	-	N	Y	Y	N	N	Y	N	N	N	N	N
SOCRADES	Y	Y	Y	N	N	N	N	Y	Y	N	N	Y	N	N	N	N	N
Thingsonomy	N	N	-	-	Y	Y	-	-	-	-	-	-	-	-	-	-	-
UBIROAD	N	N	Y	N	N	Y	N	-	-	-	-	-	-	-	-	-	-
UBISOAP	N	N	Y	N	N	N	N	-	-	-	-	-	-	-	-	-	-
UBIWARE	Y	N	-	-	-	Y	N	-	-	-	Y	-	-	-	-	-	-
WEBINOS	Y	Y	N	Y	N	N	N	Y	Y	N	Y	Y	Y	Y	N	N	N
WHEREX	N	N	-	-	Y	-	-	-	-	-	-	-	-	-	-	-	-
XMPP	Y	Y	N	N	Y	N	N	Y	Y	N	N	Y	Y	N	N	N	N

Figure 2. Summary of reviewed middleware systems and major properties

Below follows the specific details of each middleware system.

2.3 ASPIRE

ASPIRE Project (Advanced Sensors and lightweight Programmable middleware for Innovative Rfid Enterprise applications) [8] is a EU-funded project that created an open, royalty-free middleware for RFID-based applications. There is insufficient description of the security architecture to make any sensible review.

2.4 UBIWARE

The UBIWARE project is a smart semantic middleware for Ubiquitous Computing [106]. The security model for UBIWARE is not clearly described in the original paper, but an additional paper describes a model called Smart Ubiquitous Resource Privacy and Security (SURPAS) [88], which provides a security model for UBIWARE. UBIWARE is designed to utilize the semantic web constructs, and SURPAS utilises the same model of semantic web as the basis for the abstract and concrete security architectures that it proposes. The model is highly driven by policies and these can be stored and managed by external parties. In particular the SURPAS architecture is highly dynamic, allowing devices to take on board new roles or functions at runtime. While the SURPAS model describes a theoretical solution to the approach, there are few details on the concrete instantiation. For example, while the model defines a policy-based approach to access control, there are no clearly defined policy languages chosen. There is no clear model of identity or federation, and there is no clear guidance on how to ensure that federated policies that are stored on external servers are protected and maintain integrity. The model does not address any edge computing approaches or filtering/summarisation of IoT data. However, the overall approach of using ontologies and basing policies on those ontologies is very powerful.

2.5 UBIROAD

The UBIROAD middleware [114] is a specialization of the UBIWARE project specifically targeting traffic, road management, transport management and related use-cases. There is insufficient description of the security and trust architecture to make any meaningful review.

2.6 UBISOAP

ubiSOAP [24] is a Service-Oriented Architecture (SOA) approach that builds a middleware for Ubiquitous Computing and IoT based on the Web Services (WS) standards and the SOAP protocol. There is insufficient description of the security and trust architecture to make any meaningful review.

2.7 SMEPP

Secure Middleware for P2P (SMEPP) [16] is an IoT middleware explicitly designed to be secure, especially dealing with challenges in the peer-to-peer model. SMEPP security is based around the concept of a group. When a peer attempts to join a group, the system relies on challenge-response security to implement mutual authentication. At this point the newly joined peer is issues a shared session key which is shared by all members of the group. SMEPP utilizes elliptic key cryptography to reduce the burden of the security encryption onto smaller devices. Overall SMEPP has addressed security effectively for peer-to-peer groups, but assumes a wider PKI infrastructure for managing the key model used within each group. In addition, there is no discussion of access control or federated identity models, which are important for IoT scenarios. The model is that any member of the group can read data published to the group using the shared session key.

2.8 SOCRADES

SOCRADES [34] is a middleware specifically designed for manufacturing shop floors and other industrial environments. Based on SOAP and the WS stack it utilizes the security models of the WS stack, in particular the WS-Security standard for encryption and message integrity. There is no special support for federation, tokens or policy-based access control (instead relying on role-based access control). The resulting XML approach is very heavyweight for IoT devices and costly in terms of network and power [39]. In addition, the lack of explicit support for tokens and federated security and identity models creates a significant challenge in key distributions and centralized identity for this approach.

2.9 SIRENA

SIRENA (Service Infrastructure for Real-time Embedded Networked Devices) [19] is a SOAP/WS-based middleware for IoT and embedded devices. While there is little description of the security framework in SIRENA, it does show the use of the WS-Security specification. As previously discussed, this approach is very heavyweight, has issues with key distribution, federated identity and access control.

2.10 WHEREX

WhereX [49] is an event-based middleware for the IoT. There is insufficient description of the security and trust architecture to make any meaningful review.

2.11 WEBINOS

The Webinos [36] system has a well-thought through security architecture. The Webinos system is based around the core concept of devices being in the personal control of users and therefore having each user having a "personal zone" to protect. This is a more advanced concept but in the same vein as the protected sub-domains in VIRTUS. In the Webinos model, each user has a cloud instance - known as the Personal Zone Hub (PZH) that supports their devices. The Personal Zone Hub acts as a service to collect and offer access to data and capabilities of the user's devices. The PZH acts as a certificate authority, issuing certificates to the devices that are used for mutual authentication using TLS. User's authenticate to the PZH using the OpenID protocol. On the device, a communications module known as the Personal Zone Proxy (PZP) handles all communications with the PZH.

The idea of the Personal Zone may have significant issues however, when a single device is used by many different people (for example, the in-car system in a taxi as opposed to a personal vehicle). These issues are not addressed in Webinos, though they are called out in the lessons learnt.

Webinos utilizes policy-based access control modelled in the XACML [50] language. The system pushes XACML policies out to devices to limit the spread of personal and contextual data.

Webinos addresses the issue of software modification using an attestation API, which can report whether the software running is the correct level. This requires the device to be utilising Trusted Platform Module (TPM) hardware that can return attestation data.

Webinos also addresses the issue of using secure storage on devices where the device has such storage.

While the Webinos project does address many of the privacy concerns of users through the use of the Personal Zone Hub, there is clearly further work that could be done. In particular the ability for users to define what data they share with other users or other systems using a protocol such as OAuth2 [54], and the ability to install filters or other anonymising or data reduction aggregators into the PZH are lacking. One other aspect of Webinos that is worth drawing attention to is the reliance on a certain size of device: the PZP that is needed on the device is based on the *node.js* framework and therefore the device needs to be of a certain size (e.g. a 32-bit processor running a Linux derivative or similar) to participate in Webinos.

2.12 GSN
The GSN framework [2] (Global Sensor Networks) defines a middleware for the Internet of Things that requires little or no programming. The security architecture of the system is not described in any detail: there are diagrams of the container architecture which point to access control and integrity checks, but unfortunately there is not sufficient discussion to be able to categorize or evaluate the approach taken.

2.13 MOSDEN
MOSDEN (Mobile Sensor Data Processing Engine) [94] is an extension of the GSN approach (see above) which is explicitly targeted at *opportunistic* sensing from restricted devices. As with GSN, there is insufficient description of the security and trust architecture to make any meaningful review.

2.14 Thingsonomy
Thingsonomy [56] is an event-based publish-subscribe based approach that applies semantic technology and semantic matching to the events published within the system. There is no description of a security model.

2.15 OpenIoT
OpenIoT is an open cloud-based middleware for the Internet of Things. It also extends the GSN framework. There is insufficient description of the security model to make any meaningful review.

2.16 Dioptase
Dioptase [17] is a RESTful stream-processing middleware for IoT. Dioptase does address a number of useful aspects for privacy, including intermediate stream processing of data, summarisation and filtering. However, there is no detailed security architecture or description and the security model is left as an item of future work.

2.17 VIRTUS
The VIRTUS middleware [32] utilizes the core security features of the XMPP protocol to ensure security. This includes tunnelling communications over TLS, authentication via SASL, and access control via XMPP's built-in mechanisms. SASL is a flexible mechanism for authentication which supports a number of different systems including token-based approaches such as OAuth2 or Kerberos, username/password, or X.509 certificates. For client-to-server based communications, it is not clear from the description which of these methods is actually implemented within VIRTUS. For server-to-server communications there is specified the use of SASL to ensure full server federation.

While the VIRTUS model does not describe the challenges of implementing a personal instance of middleware for single users or devices, there is a concept of edge computing described, where some interactions may happen within an edge domain (e.g. within a house) and lower security is required within that domain while higher security is expected when sharing that data outside. This model is fairly briefly described but provides an interesting approach. One challenge is that there are multiple assumptions to this: firstly, that security within the limited domain needs less security, when there may be attackers within that perimeter. Secondly, that the open channel to the wider internet cannot be misused to attack the edge network. The ability to calculate, summarise and/or filter data from the edge network before sharing it is also not discussed except in very granular terms (e.g. some data are available, other data are not).

2.18 Hydra / Linksmart
Hydra [40] was a European Union funded project which has since been extended and renamed as LinkSmart. The Hydra team published a detailed theoretical model of a policy-based security approach [3].

This model is based on using lattices to define the flow of information through a system. This model provides a language-based approach to security modelling. However, whilst this paper is published as part of the Hydra funded project, there is no clear implementation of this in the context of IoT or description of how this work can benefit the IoT world. Hoever, because Hydra / Linksmart is an Open Source project [73] with documentation beyond the scientific papers, it is possible to understand the security model in greater detail by review of this project.

The Hydra and LinkSmart architectures are both based on the Web Services (WS) specifications, building on the SOAP protocol [52], which in turn builds on the XML Language [22]. The security model is described in some detail in the LinkSmart documentation [72]. The model utilises XML Security [38]. There are significant challenges in using this model in the IoT world. XML Security has a number of performance issues which are exacerbated by the need to utilise this in an IoT context. For example, any digital signature in XML Security needs a process known as XML Canonicalisation (XML C14N). XML Canonicalisation is a costly process in both time and memory. [18] shows that the memory usage is more than $10\times$ the size of the message in memory (and XML messages are already large for IoT devices). The Hydra/Linksmart approach also uses symmetric keys for security which is a challenge for IoT because each key must be uniquely created, distributed and updated upon expiry into each device creating a major key management issue.

Hydra / Linksmart offers a service called the TrustManager. This is a system that uses the cryptographic capabilities to support a trusted identity for IoT devices. This works with a Public Key Infrastructure (PKI) and certificates to ensure trust. Once again there are challenges in the distribution and management of the certificates to the devices which are not addressed in this middleware.

The Hydra middleware does not offer any policy based access control for IoT data, and does not address the secure storage of data for users, nor offer any user-controlled models of access control to user's data.

2.19 EDSOA

An Event-driven Service-oriented Architecture for the Internet of Things Service Execution [67] describes an approach that utilizes an event-driven SOA. There is no security model described.

2.20 DREMS

Distributed RealTime Managed Systems (DREMS) [70] is a combination of software tooling and a middleware runtime for IoT. It includes Linux Operating System extensions as well. DREMS is based on an actor [4] model has a well-defined security model that extends to the operating system. The security model includes the concept of multi-level security (MLS) for communications between a device and the actor. The MLS model is based on *labelled* communications. This ensures that data can only flow to systems that have a higher *clearance* than the data being transmitted. This is a very powerful security model for government and military use-cases. However, this approach does not address needs-based access control. For example, someone with *Top Secret* clearance may read data that is categorised as *Secret* even if they have no business reason to utilise that data. The weaknesses of this model have been shown with situations such as the Snowden revelations.

2.21 XMPP

The paper [59] describes how the XMPP architecture can be applied to the challenges of M2M and hence the IoT, together with a proof-of-concept approach. The system relies on the set of XMPP extensions around publish/subscribe and the related XMPP security models to implement security. This includes TLS for encryption, and access control models around publish-subscribe. There is also a discussion about leakage of information such as *presence* from devices. The proof-of-concept model did not include any federated identity models, but did utilize a One-Time Password (OTP) model on top of XMPP to address the concepts such as temporary loans of devices.

2.22 Cloud-based Car Parking Middleware

In [61] the authors describe an OSGi-based middleware for smart cities enabling IoT-based car parking. There is no description of the security architecture.

2.23 NAPS

The *Naming, Addressing and Profile Server* (NAPS) [74] describes a heterogeneous middleware for IoT based on unifying data streams from multiple IoT approaches. Based on RESTful APIs, the NAPS approach includes a key component handling Authentication, Authorization, and Accounting (AAA). The design is based on the Network Security Capability model defined in the ETSI M2M architecture [42]. However, the main details of the security architecture have not yet been implemented and have been left for future work. There is no consideration of federated identity or policy based access control.

2.24 SBIOTCM

In *A SOA Based IOT Communication Middleware* [124] is a middleware based on SOAP and WS. There is no security model described.

3 SUMMARY OF IOT MIDDLEWARE SECURITY

In reviewing both the security and privacy challenges of the wider IoT and a structured review of more than twenty middleware platforms, we have identified some key categories that can be applied across these areas.

Firstly, we must deal with the significant proportion of the systems that did not address security, left it for further work, or did not describe the security approach in any meaningful detail. This category includes WHEREX, ASPIRE, GSN, Thingsonomy, Dioptase, OpenIoT, UBIROAD, UBISOAP, CBCPM, and EDSOA.

There are two further approaches (UBIWARE, NAPS) that offer theoretical models but did not demonstrate any real-world implementation or concrete approach.

The next clear category are those middlewares that apply the SOAP/Web Services model of security. This includes SOCRADES, SIRENA, and Hydra/Linksmart. As we have discussed in the previous sections there are significant challenges in performance, memory footprint, processor power and usability of these approaches when used with the IoT.

Two of the approaches delegate the model to the XMPP standards: VIRTUS and XMPP [33, 59]. XMPP also has the complexity of XML, but avoids the major performance overheads by using TLS instead of XML Encryption and XML Security.

This finally leaves a few unique approaches, each of which brings their own unique benefits.

DREMS is the only system to provide Multi-level security based on the concept of security clearances. While this model is attractive to government and military circles (because of the classification systems used in those circles), we would argue that it fails in many regards for IoT. In particular there are no personal controls, no concept of federated identity and no policy based access controls in this model.

SMEPP offers a model based on public key infrastructures and shared session keys. We would argue this approach has a number of challenges scaling to the requirements of the IoT. Firstly, there are significant issues in key distribution and key revocation. Secondly, this model creates a new form of perimeter - based on the concept of a shared session key. That means that if one device is compromised then the data and control of all the devices in that group are also compromised.

Only Dioptase supports the concept of stream processing in the cloud, which we argue is a serious requirement for the IoT. The requirement is to be able to filter, summarise and process streams of data from devices to support anonymisation and reduction of data leakage.

Finally, the we identified that the most advanced approach is that proposed by Webinos. Webinos utilizes some key technologies to provide a security and privacy model. Firstly, this uses policy-based access control (XACML). The model does not however support user-guided access control mechanisms such as OAuth2 or UMA.

Secondly, there is the use of Federated Identity tokens (OpenID), but only from users to the cloud, as opposed to devices to the cloud. The model of using federated identity tokens from the device to the cloud is proposed by [45, 46, 30]. However, in our opinion, the main contribution of this work is the concept of Personal Zone Hub, which is a cloud service dedicated to a single user to handle the security and privacy requirements of that user. However, the PZH model from Webinos does not examine many of the further challenges of how to implement the PZH in real life. For example, user registration, cloud hosting, and many other aspects need to be defined in more detail before the Webinos PZH model is practicable for real world projects. In addition there are challenges using the PZH model with smaller devices.

3.1 Overall gaps in the middleware

When we look at the requirements for security and privacy of the Internet of Things we can see there are some gaps that are not provided by any of the reviewed middleware systems.

- None of the middleware systems explicitly applied the concept of Privacy by Design in designing a middleware directly to support privacy, although Webinos did exhibit many of the characteristics of a system that used this approach.

- None of the models applied any concepts of context-based security or reputation to IoT devices.

- None of the middleware systems offered a user-centric model of access control.

- None of the middleware systems utilised federated identity at the device level.

4 SUMMARY AND CONCLUSIONS

4.1 Contributions

In this paper we have taken a two-phase approach to reviewing the available literature around the security and privacy of IoT devices.

In the first part we created a matrix of security challenges that applied the existing CIA+ model to three distinct areas: device, network and cloud. This new model forms a clear contribution to the literature. In each of these areas we either identified that there a no distinct challenges or we identified a set of clear challenges that is either unique to or exacerbated by the IoT.

In the second part, we used a structured search approach to identity 22 specific IoT middleware frameworks and we analysed the security models of each of those. While there are existing surveys of IoT middleware, none of them focussed on a detailed analysis of the security of the surveyed systems and therefore this has a clear contribution to the literature.

4.2 Further Work

In our survey, we have identified some clear gaps. Over half the surveyed systems had either no security or no substantive discussion of security. Out of 22 surveyed systems we found very few that addressed a significant proportion of the major challenges that we identified in the first section. We found certain aspects that were identified in the first section that were not addressed by any of the surveyed systems. Based on this we believe there is a significant opportunity to contribute to the research by creating a middleware for IoT that addresses these gaps.

- To define a model and architecture for IoT middleware that is designed from the start to enable privacy and security (Privacy by Design).

- Secondly, to bring together the best practice into a single middleware that includes: federated identity (for users and devices), policy-based access control, user managed access to data, stream processing in the cloud.

- Thirdly, there is considerable work to be done to define a better model around the implementation challenges for the concept of a personal cloud service (e.g the Webinos PZH). This includes the hosting model, bootstrapping, discovery and usage for smaller devices.

- Finally, creating a middleware system that applies context-based security and reputation to IoT middleware.

REFERENCES

[1] ABDUL-RAHMAN, A., AND HAILES, S. Supporting trust in virtual communities. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6* (Washington, DC, USA, 2000), IEEE Computer Society.

[2] ABERER, K., AND HAUSWIRTH, M. Middleware support for the" internet of things".

[3] ADETOYE, A. O., AND BADII, A. Foundations and applications of security analysis. Springer-Verlag, Berlin, Heidelberg, 2009, ch. A Policy Model for Secure Information Flow, pp. 1–17.

[4] AGHA, G. A. Actors: A model of concurrent computation in distributed systems. Tech. rep., DTIC Document, 1985.

[5] ANDERSSON, K., AND SZEWCZYK, P. Insecurity by obscurity continues: are adsl router manuals putting end-users at risk.

[6] ARDUINO. Arduino. `http://arduino.cc/`, 2015.

[7] ASHTON, K. That 'internet of things' thing. *RFiD Journal 22* (2009), 97–114.

[8] ASPIRE, F. Fp7 ict ip project advanced sensors and lightweight programmable middleware for innovative rfid enterprise applications (aspire), 2008.

[9] ATMEL. Master's thesis, 2015.

[10] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer networks 54*, 15 (2010), 2787–2805.

[11] AUDET, F., AND JENNINGS, C. Network address translation (nat) behavioral requirements for unicast udp. Tech. rep., 2007.

[12] AUGUSTO, A. B., AND CORREIA, M. E. An xmpp messaging infrastructure for a mobile held security identity wallet of personal and private dynamic identity attributes. *Proceedings of the XATA* (2011).

[13] BALL, J. Nsa stores metadata of millions of web users for up to a year, secret files show. `http://www.theguardian.com/world/2013/sep/30/nsa-americans-metadata-year-documents`, 2013. (Visited on 06/08/2015).

[14] BANDYOPADHYAY, S., SENGUPTA, M., MAITI, S., AND DUTTA, S. Role of middleware for internet of things: A study. *International Journal of Computer Science & Engineering Survey (IJCSES) 2*, 3 (2011), 94–105.

[15] BANDYOPADHYAY, S., SENGUPTA, M., MAITI, S., AND DUTTA, S. A survey of middleware for internet of things. In *Recent Trends in Wireless and Mobile Networks*. Springer, 2011, pp. 288–296.

[16] BENITO, R. J. C., MÁRQUEZ, D. G., TRON, P. P., CASTRO, R. R., MARTÍN, N. S., AND MARTÍN, J. L. S. Smepp: A secure middleware for embedded p2p. *Proceedings of ICT-MobileSummit 9* (2009).

[17] BILLET, B., AND ISSARNY, V. Dioptase: a distributed data streaming middleware for the future web of things. *Journal of Internet Services and Applications 5*, 1 (2014), 1–19.

[18] BINNA, M. www.w3.org/2008/xmlsec/papers/c14n2_performance_evaluation_thesis.pdf. `http://www.w3.org/2008/xmlsec/papers/C14N2_Performance_Evaluation_Thesis.pdf`, 2008. (Visited on 06/09/2015).

[19] BOHN, H., BOBEK, A., AND GOLATOWSKI, F. Sirena-service infrastructure for real-time embedded networked devices: A service oriented framework for different domains. In *Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006. ICN/ICONS/MCL 2006. International Conference on* (2006), IEEE, pp. 43–43.

[20] BOJINOV, H., MICHALEVSKY, Y., NAKIBLY, G., AND BONEH, D. Mobile device identification via sensor fingerprinting. *arXiv preprint arXiv:1408.1416* (2014).

[21] BOTEZATU, B. 25 percent of wireless networks are highly vulnerable to hacking attacks, wi-fi security survey reveals — hotforsecurity. `http://www.hotforsecurity.com/blog/25-percent-of-wireless-networks-are-highly-vulnerable-to-hacking-attacks-w.html`, 2011. (Visited on 07/14/2015).

[22] BRAY, T. E. A. Extensible Markup Language (XML) 1.0. Recommendation, W3C, February 2004. Available at `http://www.w3.org/TR/REC-xml`.

[23] BRICKELL, E., CAMENISCH, J., AND CHEN, L. Direct anonymous attestation. In *Proceedings of the 11th ACM conference on Computer and communications security* (2004), ACM, pp. 132–145.

[24] CAPORUSCIO, M., RAVERDY, P.-G., AND ISSARNY, V. ubisoap: A service-oriented middleware for ubiquitous networking. *Services Computing, IEEE Transactions on 5*, 1 (2012), 86–98.

[25] CARD, J. Anonymity is the internet's next big battleground. http://www.theguardian.com/media-network/2015/jun/22/ anonymity-internet-battleground-data-advertisers-marketers, 2015. (Visited on 07/13/2015).

[26] CAVOUKIAN, A. Privacy in the clouds. *Identity in the Information Society 1*, 1 (2008), 89–108.

[27] CHAKRAVORTY, R., CARTWRIGHT, J., AND PRATT, I. Practical experience with tcp over gprs. In *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE* (2002), vol. 2, IEEE, pp. 1678–1682.

[28] CHAQFEH, M., MOHAMED, N., ET AL. Challenges in middleware solutions for the internet of things. In *Collaboration Technologies and Systems (CTS), 2012 International Conference on* (2012), IEEE, pp. 21–26.

[29] CHEN, D., CHANG, G., SUN, D., LI, J., JIA, J., AND WANG, X. Trm-iot: A trust management model based on fuzzy reputation for internet of things. *Computer Science and Information Systems 8*, 4 (2011), 1207–1228.

[30] CIRANI, S., PICONE, M., GONIZZI, P., VELTRI, L., AND FERRARI, G. IoT-OAS: An OAuth-based Authorization Service Architecture for Secure Services in IoT Scenarios.

[31] CLUSTER CERP-IoT. Internet of things, strategic research roadmap. *European Commission* (2009).

[32] CONZON, D., BOLOGNESI, T., BRIZZI, P., LOTITO, A., TOMASI, R., SPIRITO, M., ET AL. The virtus middleware: An xmpp based architecture for secure iot communications. In *Computer Communications and Networks (ICCCN), 2012 21st International Conference on* (2012), IEEE, pp. 1–6.

[33] CONZON, D., BOLOGNESI, T., BRIZZI, P., LOTITO, A., TOMASI, R., SPIRITO, M., ET AL. The virtus middleware: An xmpp based architecture for secure iot communications. In *Computer Communications and Networks (ICCCN), 2012 21st International Conference on* (2012), IEEE, pp. 1–6.

[34] DE SOUZA, L. M. S., SPIESS, P., GUINARD, D., KÖHLER, M., KARNOUSKOS, S., AND SAVIO, D. Socrades: A web service based shop floor integration infrastructure. In *The internet of things*. Springer, 2008, pp. 50–67.

[35] DEITEL, H. M. *An introduction to operating systems*, vol. 3. Addison-Wesley Reading, Massachusetts, 1984.

[36] DESRUELLE, H., LYLE, J., ISENBERG, S., AND GIELEN, F. On the challenges of building a web-based ubiquitous application platform. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (2012), ACM, pp. 733–736.

[37] DIERKS, T. The transport layer security (tls) protocol version 1.2.

[38] DOURNAEE, B., AND DOURNEE, B. *XML security*. Mcgraw-hill, 2002.

[39] DUNKELS, A., ET AL. Efficient application integration in ip-based sensor networks. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings* (2009), ACM, pp. 43–48.

[40] EISENHAUER, M., ROSENGREN, P., AND ANTOLIN, P. A development platform for integrating wireless devices and sensors into ambient intelligence systems. In *Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2009. SECON Workshops' 09. 6th Annual IEEE Communications Society Conference on* (2009), IEEE, pp. 1–3.

[41] ET AL, T. Authentication and authorization for constrained environments using oauth and uma.

[42] ETSI. Etsi - m2m. http://www.etsi.org/technologies-clusters/ technologies/m2m, 2015. (Visited on 07/08/2015).

[43] EVANS, D. The internet of things. *How the Next Evolution of the Internet is Changing Everything, Whitepaper, Cisco Internet Business Solutions Group (IBSG)* (2011).

[44] FITBIT. Fitbit official site for activity trackers & more. http://www.fitbit.com/, 2015. (Visited on 07/09/2015).

[45] FREMANTLE, P., AZIZ, B., SCOTT, P., AND KOPECKY, J. Federated Identity and Access Management for the Internet of Things. In *3rd International Workshop on the Secure IoT* (2014).

[46] FREMANTLE, P., KOPECKÝ, J., AND AZIZ, B. Web api management meets the internet of things. In *Services and Applications over Linked APIs and Data – SALAD2015* (2015).

[47] FULLAM, K., AND BARBER, K. Learning trust strategies in reputation exchange networks. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems* (2006), ACM Press, pp. 1241–1248.

[48] FURBER, S. B. *ARM system Architecture*. Addison-Wesley Longman Publishing Co., Inc., 1996.

[49] GIUSTO, D., IERA, A., MORABITO, G., AND ATZORI, L. *The internet of things: 20th Tyrrhenian workshop on digital communications*. Springer Science & Business Media, 2010.

[50] GODIK, S., ANDERSON, A., PARDUCCI, B., HUMENN, P., AND VAJJHALA, S. Oasis extensible access control 2 markup language (xacml) 3. Tech. rep., Tech. rep., OASIS, 2002.

[51] GOODIN, D. New linux worm targets routers, cameras, internet of things devices, 2013.

[52] GUDGIN, M. E. A. SOAP Version 1.2 Part 1: Messaging Framework. Recommendation, W3C, June 2003. Available at http://www.w3.org/TR/2003/REC-soap12-part1-20030624/.

[53] GURA, N., PATEL, A., WANDER, A., EBERLE, H., AND SHANTZ, S. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *Cryptographic Hardware and Embedded Systems - CHES 2004*, M. Joye and J.-J. Quisquater, Eds., vol. 3156 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004, pp. 119–132.

[54] HAMMER-LAHAV, D., AND HARDT, D. The oauth2.0 authorization protocol. 2011. Tech. rep., IETF Internet Draft, 2011.

[55] HANKS, P. Collins dictionary of the english language. *London: Collins,— c1986, 2nd ed., edited by Hanks, Patrick 1* (1986).

[56] HASAN, S., AND CURRY, E. Thingsonomy: Tackling variety in internet of things events. *Internet Computing, IEEE 19*, 2 (2015), 10–18.

[57] HERNÁNDEZ-RAMOS, J. L., JARA, A. J., MARIN, L., AND SKARMETA, A. F. Distributed capability-based access control for the internet of things. *Journal of Internet Services and Information Security (JISIS) 3*, 3/4 (2013), 1–16.

[58] HILL, K. When 'smart homes' get hacked: I haunted a complete stranger's house via the internet - forbes. http://www.forbes.com/sites/kashmirhill/2013/07/26/smart-homes-hack/, 2013. (Visited on 07/09/2015).

[59] IIVARI, A., VÄISÄNEN, T., BEN ALAYA, M., RIIPINEN, T., AND MONTEIL, T. Harnessing xmpp for machine-to-machine communications & pervasive applications. *Journal of Communications Software & Systems 10*, 3 (2014).

[60] INITIATIVE, K., ET AL. User managed access (uma), 2013.

[61] JI, Z., GANCHEV, I., O'DROMA, M., ZHAO, L., AND ZHANG, X. A cloud-based car parking middleware for iot-based smart cities: design and implementation. *Sensors 14*, 12 (2014), 22372–22393.

[62] JØSANG, A., ISMAIL, R., AND BOYD, C. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems 43*, 2 (March 2007), 618–644.

[63] KEOH, S., KUMAR, S., AND GARCIA-MORCHON, O. Securing the ip-based internet of things with dtls. *Working Draft, February* (2013).

[64] KHURANA, H., HADLEY, M., LU, N., AND FRINCKE, D. A. Smart-grid security issues. *Security & Privacy, IEEE 8*, 1 (2010), 81–85.

[65] KOBLITZ, N. Elliptic curve cryptosystems. *Mathematics of computation 48*, 177 (1987), 203–209.

[66] KOSCHUCH, M., HUDLER, M., AND KRÜGER, M. Performance evaluation of the tls handshake in the context of embedded devices. In *Data Communication Networking (DCNET), Proceedings of the 2010 International Conference on* (2010), IEEE, pp. 1–10.

[67] LAN, L., WANG, B., ZHANG, L., SHI, R., AND LI, F. An event-driven service-oriented architecture for internet of things service execution. *International Journal of Online Engineering (iJOE) 11*, 2 (2015), pp–4.

[68] LANDMAN, D. Davylandman/aeslib. `https://github.com/DavyLandman/AESLib`, 2015. (Visited on 07/09/2015).

[69] LARSON, J., PERLROTH, N., AND SHANE, S. The nsa's secret campaign to crack, undermine internet encryption - propublica. `http://www.propublica.org/article/the-nsas-secret-campaign-to-crack-undermine-internet-encryption`, 2013. (Visited on 06/08/2015).

[70] LEVENDOVSZKY, T., DUBEY, A., OTTE, W. R., BALASUBRAMANIAN, D., COGLIO, A., NYAKO, S., EMFINGER, W., KUMAR, P., GOKHALE, A., AND KARSAI, G. Distributed real-time managed systems: A model-driven distributed secure information architecture platform for managed embedded systems. *Software, IEEE 31*, 2 (2014), 62–69.

[71] LEVINSON, L. Secrets, lies and snowden's email: why i was forced to shut down lavabit. `http://www.theguardian.com/commentisfree/2014/may/20/why-did-lavabit-shut-down-snowden-email`, 2014. (Visited on 06/08/2015).

[72] LINKSMART. Eulinksmartsecuritycommunicationsecuritymanagersym - linksmart open source middleware - linksmart middleware portal. `https://linksmart.eu/redmine/projects/linksmart-opensource/wiki/Eulinksmartsecuritycommunicationsecuritymanagersym`, 2015. (Visited on 06/09/2015).

[73] LINKSMART.EU. Linksmart middleware portal. `https://linksmart.eu/redmine`, 2015. (Visited on 07/09/2015).

[74] LIU, C. H., YANG, B., AND LIU, T. Efficient naming, addressing and profile services in internet-of-things sensory environments. *Ad Hoc Networks 18* (2014), 85–101.

[75] LOCKE, D. Mq telemetry transport (mqtt) v3. 1 protocol specification. *IBM developerWorks Technical Library], available at http://www. ibm. com/developerworks/webservices/library/ws-mqtt/index. html* (2010).

[76] LOMNE, V., DEHABOUI, A., MAURINE, P., TORRES, L., AND ROBERT, M. Side channel attacks. In *Security Trends for FPGAS*. Springer, 2011, pp. 47–72.

[77] LUCKENBACH, T., GOBER, P., ARBANOWSKI, S., KOTSOPOULOS, A., AND KIM, K. Tinyrest-a protocol for integrating sensor networks into the internet. In *Proc. of REALWSN* (2005), pp. 101–105.

[78] MAHALLE, P. N., ANGGOROJATI, B., PRASAD, N. R., AND PRASAD, R. Identity establishment and capability based access control (iecac) scheme for internet of things. In *Wireless Personal Multimedia Communications (WPMC), 2012 15th International Symposium on* (2012), IEEE, pp. 187–191.

[79] MCDANIEL, P., AND MCLAUGHLIN, S. Security and privacy challenges in the smart grid. *Security & Privacy, IEEE 7*, 3 (2009), 75–77.

[80] MICHIARDI, P., AND MOLVA, R. Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Advanced Communications and Multimedia Security*. Springer, 2002, pp. 107–121.

[81] MILLER, V. S. Use of elliptic curves in cryptography. In *Advances in Cryptology—CRYPTO'85 Proceedings* (1986), Springer, pp. 417–426.

[82] MONTANARI, R., TONINELLI, A., AND BRADSHAW, J. M. Context-based security management for multi-agent systems. In *Multi-Agent Security and Survivability, 2005 IEEE 2nd Symposium on* (2005), IEEE, pp. 75–84.

[83] MORRIS, T. Trusted platform module., 2011.

[84] MOSKOWITZ, R. Hip diet exchange (dex).

[85] MOSKOWITZ, R. Host identity protocol architecture.

[86] MURPHY, C. Internet of things: Who gets the data? - informationweek. http://www.informationweek.com/strategic-cio/executive-insights-and-innovation/internet-of-things-who-gets-the-data/a/d-id/1252701, 2014. (Visited on 06/08/2015).

[87] NARAYANAN, A., AND SHMATIKOV, V. Robust de-anonymization of large sparse datasets. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on* (2008), IEEE, pp. 111–125.

[88] NAUMENKO, A., KATASONOV, A., AND TERZIYAN, V. A security framework for smart ubiquitous industrial resources. In *Enterprise Interoperability II*. Springer, 2007, pp. 183–194.

[89] NEWSOME, J., SHI, E., SONG, D., AND PERRIG, A. The sybil attack in sensor networks: analysis & defenses. In *Proceedings of the 3rd international symposium on Information processing in sensor networks* (2004), ACM, pp. 259–268.

[90] O. GARCIA-MORCHON, E. A. Security Considerations in the IP-based Internet of Things. Internet Draft, IETF, September 2013. Available at http://tools.ietf.org/html/draft-garcia-core-security-06.

[91] PARK, K.-W., SEOK, H., AND PARK, K.-H. pkasso: towards seamless authentication providing non-repudiation on resource-constrained devices. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on* (2007), vol. 2, IEEE, pp. 105–112.

[92] PAVERD, A., AND MARTIN, A. Hardware security for device authentication in the smart grid. In *First Open EIT ICT Labs Workshop on Smart Grid Security - SmartGridSec12* (Berlin, Germany, 2012).

[93] PERELMAN, V., AND ERSUE, M. Tls with psk for constrained devices.

[94] PERERA, C., JAYARAMAN, P. P., ZASLAVSKY, A., GEORGAKOPOULOS, D., AND CHRISTEN, P. Mosden: An internet of things middleware for resource constrained mobile devices. In *System Sciences (HICSS), 2014 47th Hawaii International Conference on* (2014), IEEE, pp. 1053–1062.

[95] PERRIG, A., SZEWCZYK, R., TYGAR, J., WEN, V., AND CULLER, D. E. Spins: Security protocols for sensor networks. *Wireless networks 8*, 5 (2002), 521–534.

[96] PFLEEGER, C. P., AND PFLEEGER, S. L. *Security in computing*. Prentice Hall Professional Technical Reference, 2002.

[97] RADOMIROVIC, S. Towards a model for security and privacy in the internet of things. In *Proc. First Int'l Workshop on Security of the Internet of Things* (2010).

[98] RENDLE, A. Who owns the data in the internet of things? http://www.taylorwessing.com/download/article_data_lot.html, 2014. (Visited on 06/08/2015).

[99] RESCORLA, E., AND MODADUGU, N. Datagram transport layer security. Tech. rep., 2006.

[100] RIVEST, R. L., SHAMIR, A., AND ADLEMAN, L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM 21*, 2 (1978), 120–126.

[101] RYAN, M. Bluetooth: With low energy comes low security. In *WOOT* (2013).

[102] SADEGHI, A.-R., AND STÜBLE, C. Property-based attestation for computing platforms: caring about properties, not mechanisms. In *Proceedings of the 2004 workshop on New security paradigms* (2004), ACM, pp. 67–77.

[103] SAINT-ANDRE, P. Extensible messaging and presence protocol (xmpp): Core.

[104] SAKIMURA, N., BRADLEY, J., AND JONES, M. Openid connect dynamic client registration 1.0-draft 14, 2013.

[105] SAMSUNG. Mobile Enterprise Security — Samsung KNOX. https://www.samsungknox.com/en, 2015. (Visited on 03/24/2015).

[106] Scuturici, V.-M., Surdu, S., Gripay, Y., and Petit, J.-M. Ubiware: Web-based dynamic data & service management platform for ami. In *Proceedings of the Posters and Demo Track* (2012), ACM, p. 11.

[107] Seshadri, A., Perrig, A., Van Doorn, L., and Khosla, P. Swatt: Software-based attestation for embedded devices. In *Security and Privacy, 2004. Proceedings. 2004 IEEE Symposium on* (2004), IEEE, pp. 272–282.

[108] Sethi, M. Security in smart object networks. Master's thesis, Aalto University, School of Science, 2012.

[109] Sethi, M., Arkko, J., and Keranen, A. End-to-end security for sleepy smart object networks. In *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on* (2012), IEEE, pp. 964–972.

[110] Silaghi, G. C., Arenas, A., and Silva, L. M. Reputation-based trust management systems and their applicability to grids. Tech. Rep. TR-0064, Institutes on Knowledge and Data Management and System Architecture, CoreGRID - Network of Excellence, February 2007.

[111] Simmonds, A., Sandilands, P., and Van Ekert, L. An ontology for network security attacks. In *Applied Computing*. Springer, 2004, pp. 317–323.

[112] Skorobogatov, S. P. *Semi-invasive attacks: a new approach to hardware security analysis*. PhD thesis, Citeseer, 2005.

[113] TCG. Trusted computing group - home. `http://www.trustedcomputinggroup.org/`, 2015. (Visited on 06/08/2015).

[114] Terziyan, V., Kaykova, O., and Zhovtobryukh, D. Ubiroad: Semantic middleware for context-aware smart road environments. In *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on* (2010), IEEE, pp. 295–302.

[115] Tschofenig, H., and Fossati, T. A tls/dtls 1.2 profile for the internet of things. *draft-ietf-dice-profile-08 (work in progress)* (2014).

[116] Turkmen, F., and Crispo, B. Performance evaluation of xacml pdp implementations. In *Proceedings of the 2008 ACM workshop on Secure web services* (2008), ACM, pp. 37–44.

[117] University of Portsmouth Library. Discovery service. `http://www.port.ac.uk/library/infores/discovery/filetodownload,170883,en.xls`, 2015. (Visited on 07/14/2015).

[118] Vieira, M. A. M., Coelho Jr, C. N., da Silva Jr, D. C., and da Mata, J. M. Survey on wireless sensor network devices. In *Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA'03. IEEE Conference* (2003), vol. 1, IEEE, pp. 537–544.

[119] Watro, R., Kong, D., Cuti, S.-f., Gardiner, C., Lynn, C., and Kruus, P. Tinypk: securing sensor networks with public key technology. In *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks* (2004), ACM, pp. 59–64.

[120] Winter, J. S. Privacy and the emerging internet of things: using the framework of contextual integrity to inform policy. In *Pacific Telecommunication Council Conference Proceedings 2012* (2012).

[121] Yan, S. Y. Side-channel attacks. In *Cryptanalytic Attacks on RSA*. Springer, 2008, pp. 207–222.

[122] Yi, S., and Kravets, R. Key management for heterogeneous ad hoc wireless networks. In *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on* (2002), IEEE, pp. 202–203.

[123] Zee. Fitbit users are unwittingly sharing details of their sex lives with the world, 2011. (Visited on 06/04/2013).

[124] Zhiliang, W., Yi, Y., Lu, W., and Wei, W. A soa based iot communication middleware. In *Mechatronic Science, Electric Engineering and Computer (MEC), 2011 International Conference on* (2011), IEEE, pp. 2555–2558.

[125] ZOURIDAKI, C., MARK, B. L., HEJMO, M., AND THOMAS, R. K. Hermes: A quantitative trust establishment framework for reliable data packet delivery in manets. *Journal of Computer Security 15*, 1 (2007), 3–38.

[126] ZOURIDAKI, C., MARK, B. L., HEJMO, M., AND THOMAS, R. K. E-hermes: A robust cooperative trust establishment scheme for mobile ad hoc networks. *Ad Hoc Networks 7*, 6 (2009), 1156–1168.