COMPUTERS & SECURITY 78 (2018) 281–300

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Tangible security: Survey of methods supporting secure ad-hoc connects of edge devices with physical context

Qiao Hu[a], Jingyi Zhang[a], Aikaterini Mitrokotsa[b], Gerhard Hancke[a,*]

[a] Department of Computer Science, City University of Hong Kong, Hong Kong
[b] Department of Computer Science and Engineering, Chalmers University of Technology, Sweden

ARTICLE INFO

Article history:
Received 19 March 2018
Revised 6 June 2018
Accepted 30 June 2018
Available online 30 July 2018

Keywords:
Physical-context security
Key management
Device pairing
Proof-of-proximity
Relay attack

ABSTRACT

Edge computing is the concept of moving computation back to the endpoints of a network, as an alternative to, or in combination with, centralized, cloud-based architectures. It is especially of interest for Internet-of-Things and Cyber-Physical Systems where embedded endpoints make up the edge of the network, and where these devices need to make localised, time-critical decisions. In these environment secure, ad-hoc device-to-device interaction is important, but offers a challenge because devices might belong to different systems, or security domains, which complicates trusted communication and key establishment. There has been a growing interest in complementing conventional cryptography with physical context. This allows for services that are difficult to achieve with existing cryptographic mechanisms: devices pairing (initial key establishment) and proof-of-proximity (ensuring devices are physically present). Numerous methods, the majority of which are based on the physical context of device characteristics, behavior or environment, have been proposed to supplement cryptography in achieving these services. This paper provides an overview of this area of research, first discussing the nature and importance of the two specified security services in ad-hoc communication settings and then providing an introduction to prominent physical context security approaches in literature.

1. Introduction

The Internet-of-Things (IoT) consists of a large number of interconnected pervasive devices, collecting data and interacting with their environment. The IoT can support many useful applications in consumer and industrial settings and it is estimated that 25 billion things will be connected to the Internet by 2020 (Technology, 2017). Simply managing each device's function centrally would require the transmission of a huge amount of data, requiring large data centers and significant network infrastructure, which would result in increased latency and degradation of quality of service. To overcome this problem, edge computing is proposed to move some centralized functions of the cloud to network endpoints. Most of the IoT 'things' are inherently functioning as network endpoints, or edge devices (Dolui and Datta, 2017). Utilizing the idle storage and processing capacities of these devices to perform tasks meant for the cloud can effectively reduce the network latency as these tasks are completed locally. Edge computing in mobile environments can either have a static or ad-hoc network structure. In this paper, we limit our research to ad-hoc connections and consider primarily wireless

* Corresponding author.
E-mail address: ghancke@ieee.org (G. Hancke).

https://doi.org/10.1016/j.cose.2018.06.009
0167-4048/

communication, such as cellular network, Wi-Fi, WiMAX, Bluetooth and Zigbee.

Connections and networked nodes in enterprise networks are carefully managed and use traditional cryptographic mechanisms to establish trusted communication links. However, for ad-hoc connections between edge devices, there are some problems that are not easily solvable using conventional cryptography. Firstly, in most cases cryptographic mechanisms require devices to share a secret symmetric key. Distributing and managing such keys in an open, ad-hoc operational environment is a very difficult problem, especially when establishing a key between two devices with no prior relationship. Public key cryptography offers a potential alternative but is commonly believed not to be a feasible solution for resource-limited devices. For example, in a tactical cloudlet (Echeverria et al., 2016), resource-limited devices are working in disconnected, intermittent and limited environments where pre-shared credentials and trusted third parties are all unrealistic. Secondly, proof-of-proximity is also becoming an important security service that tries to verify the physical 'closeness' of the communicating devices. This is important in edge computing systems that provide context-aware services based on the location of devices. For example, the multi-access edge computing platform provided by Nokia only facilitates the running of applications satisfying the location requirement. But in a network world, we cannot simply make a secure assumption that all devices are honest and an attacker could in practice simply relay messages between the end devices and edge computing nodes that are in fact not present. This ability to relay communication means that traditional cryptography is ineffective, as any cryptographic protocol exchange can simply be relayed between the two legitimate devices. These two issues are discussed in more detail in Section 2.

In looking for ways to support key pairing and proximity verification in ad-hoc environments, an interesting body of work has developed around the notion of using physical context as security mechanisms. In a way, this applies a basic principle inherent to trust being established – a tangible connection. If we were to consider a human equivalent, one person talking to another in person and physically observing the other party has a more secure, and an easier verifiable, 'connection' than if the same two people had a telephone conversation or exchange of letters. In the latter two, additional measures are needed to establish the same level of trust. Physical context mechanisms are not meant to be an alternative to existing cryptographic mechanisms but can play a supporting role in creating more secure systems. Essentially, it could be used to initially exchange a key or ensure that communication is not being relayed by an attacker.

This paper serves as a starting point for understanding the potential security services physical context can provide and the different approaches that can be used to implement these services that can be utilized to enhance the security of edge computing. We aim to specifically provide an overview of existing work on key pairing and proof-of-proximity mechanisms. The scope of our survey is limited to approaches that are (1) targeted at pervasive devices and (2) have a core focus on physical context. We present current work classified under three broad categories: mechanisms based on the devices' environment, the communication channel and the devices' physical characteristics. This topic is of potential interest to anyone looking at alternative security solutions for emerging connectivity concepts, such as the Internet-of-Things, and also contributes to the larger areas of cryptographic pairing and key management, as well as physical layer security.

2. Background

Although we can use any cryptographic security mechanisms there are some problems that are not easily solvable using conventional cryptography. The first is device pairing in a wireless environment, i.e. establishing a key between two devices with no prior relationship, and the second is proof-of-proximity, i.e. ensure that two communicating devices are physically close together.

2.1. *Key pairing*

Key management is not a straightforward process in any system. Even in systems where the interaction between devices can be planned for security, they also need to distribute long-term secret keys. If two devices need to securely communicate they need to share a key. This key can either be pre-distributed and stored by the two devices, or the devices could store a key to communicate with a central key server, which will use this key to share a short term session key with the two devices that need to communicate. Both these models are based on the assumption that both devices are managed by the same entity or have some form of relationship before communication takes place. In an open ad-hoc environment, devices might not have co-operating managing entities and it is unlikely that devices will have a pre-distributed shared key, i.e. a device cannot store a key to talk to every single other device and does not know in advance who it might interact with. An alternative is to use public-key cryptography, where a device is pre-issued a private-public key pair. It keeps the private key secret but can share the public key with any device it encounters. Any device receiving the public key can encrypt a message, but only the device with the corresponding private key can decrypt. This could be used to exchange a shared symmetric key, but if done naively could have adverse security implications and secure pairing based on public cryptography is an active research topic in itself (Mirzadeh et al., 2014). At the same time, public-key cryptography approaches are more computational intensive and are generally considered to not be practical for many pervasive devices, such as simple sensor nodes.

Key pairing based on physical context aims to provide a method for ad-hoc exchange or derivation of a shared secret key. This method should not rely on an online trusted third party being available or any prior secure relationship between the devices. We use a vehicle-assisted data delivery scheme (Cheng et al., 2016b) as an example. This scheme utilizes the vehicle to transfer data over long distances with the help of vehicle-to-vehicle communication. Manual out-of-band (OOB) pairing schemes, such as the PIN used with Bluetooth, could be used. However, this is not ideal as drivers must focus on driving. Also it is not suitable for machine-to-machine or user-to-devices communication scenarios. A

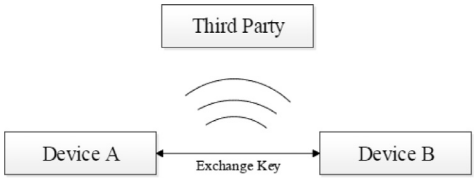

Fig. 1 – Key exchange pairing.

better solution would be invisible to the users and allow the devices to autonomously negotiate a key without human interaction. This must be possible even if an adversary is present and is able to listen to the communication of the devices, as in Fig. 1. In the literature, there are currently four general approaches to this form of key management:

- The devices could use a 'location-limited' auxiliary channel to transmit a key. The assumption is that only the intended recipient will receive the key, whereas an adversary that is further away could not recover the key. For example, using a line-of-sight optical channel, a displayed image or infra-red (Saxena et al., 2011a).
- The devices could exchange a key and use intentional channel manipulation, or 'friendly' jamming, to prevent the adversary from recovering the key. This basic approach requires both devices to transmit at the same time. The adversary receives the combined signal and is unable to determine the data sent, while each legitimate device can cancel their own transmission from the received signal and determine the data that the other device has send (Castelluccia and Avoine, 2006).
- The devices participate in a common activity during which they record their actions (Castelluccia and Mutaf, 2005). Both devices should perform the same actions, the physical measurements of these actions can be used by both devices to derive a common key. For example, two mobile devices can sample their accelerometers while being shaken together. As they have been moved in a similar way during this activity they have a common set of measurements that could be used to derive the same key on both devices.
- The devices can observe their surroundings and derive a key based on this observation. If both devices are in the same location they will theoretically observe the same environmental features and these could then be used to derive a common key. For example, two devices can listen to the ambient audio (Schurmann and Sigg, 2013) or time-varying wireless environment (Mathur et al., 2011) at their location.
- Two devices communicate through a wireless channel and they can use common channel parameters to derive a shared key (Zhang et al., 2016). Channel parameters consist of Channel Statement Information (CSI) and Received Signal Strength (RSS). The randomness of the noisy channel is the resource of secret keys.

2.2. *Proof of (Physical) proximity*

Physical proximity is important in a number of applications, e.g. devices are divided in regions or physical groups (Cheng et al., 2016a). In Advanced Metering Infrastructure (AMI), smart meters are used to record energy consumption information while data concentrator units (DCU) can collect information from smart meters. Energy allocation is made depending on these data. If smart meters in an area record faulty data, i.e. fake energy consumption, large amounts of energy would be distributed to this area, which may cause energy wastage in this area and energy shortage in other areas. A relay attack can be used to launch such attack (Fig. 2). In a relay attack an adversary control two entities - the proxy smart meter and the proxy DCU. The proxy smart meter captures the DCU's authentication challenge and relays it to the proxy DCU. The proxy DCU will challenge the legitimate smart meter using the relayed challenge and observe the legitimate authentication response generated by the smart meter. The authentication response is then relayed back to the proxy DCU. At this stage, the DCU is still awaiting a reply to its original challenge and the proxy smart meter responds with the relayed legitimate response. As the DCU receives a legitimate response from a device within its communication range, the DCU is convinced that the legitimate smart meter is present although in reality this smart meter is far away. This essentially allows the adversary to temporarily possess a 'virtual clone' of a smart meter for as long as he can continue relaying the subsequent communication.

It should be emphasized that a relay attack is not like an ordinary man-in-the-middle attack, which needs some form of security vulnerability in the protocol or requires the attacker to actively modify the data between the participants. Conventional cryptographic authentication mechanisms, regardless of the protocol or constituent cryptographic primitives, fail to detect a relay attack as any challenge and response could simply be relayed between the participants. The attacker does not need to modify, or even understand, the data that he is relaying. Implementing relay-resistant authentication, therefore, requires the use of additional, potentially non-cryptographic, mechanisms to support conventional cryptographic algorithms. In the literature, there are currently four general approaches to detecting relay attacks:

- The DCU could use multiple channels when authenticating the smart meter, thereby making it computationally harder for the adversary to relay all the channel data. For example, using multiple radio channels or using a radio channel and an audio channel (Stajano et al., 2010).
- The DCU and the smart meter could generate a 'proximity proof' by observing their environment and verifying that they both made the same observation. For example, the smart meter includes its location or observed environment in the message to the DCU, which compares it to its own location (Hu et al., 2003).
- The DCU tries to detect the additional delay the adversary introduces when relaying the messages. For example, the DCU uses distance-bounding protocols together with special channels, such as a near-field, bit-exchange channel (Hancke, 2011), to measure the round-trip time of the challenge/response exchange (in this research community relay attacks are termed 'mafia fraud') (Avoine et al., 2011).
- The DCU observes the physical properties of the device it is communicating with and uses these properties to

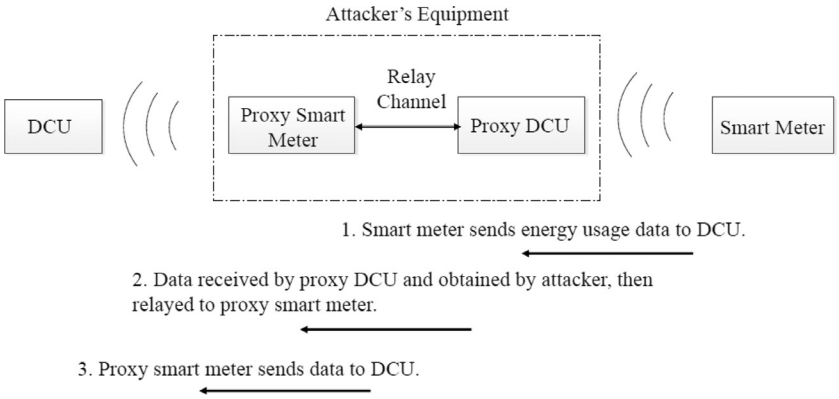

Fig. 2 – Relay attack on RFID systems.

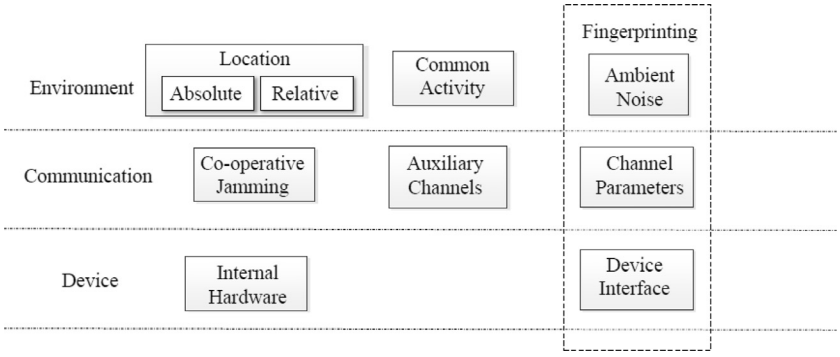

Fig. 3 – An overview of approaches to using physical context within security services.

confirm the identity of the device. In other words, it tries to differentiate between the legitimate smart meter and proxy smart meter by observing the intrinsic and inherent physical properties of the communicating device and comparing it to the known physical features of the legitimate smart meter. The features of the device itself, such as the intrinsic hardware properties (Holcomb et al., 2009; Kohno et al., 2005), physical unclonable functions (PUFs) (Ruhrmair and van Dijk, 2013), or the features of the physical-layer communication (Danev et al., 2012), could be used.

2.3. *Security schemes using physical context*

From the previous two sections, it as apparent that some of the general approaches to providing ad-hoc key pairing, and all the approaches providing relay-resistant communication, use physical context to support the overarching security services.This paper serves as an introduction and overview of how physical context is used to provide key pairing and relay resistance in an AMI system with low-resource devices. As shown in Fig, 3, physical context methods can be divided into three categories based on the source of the physical properties.

Environmental methods rely on the physical properties external to the device and its communication. This category covers both the properties of the devices' surroundings and also external actions affecting the devices. The physical location of each device could be used as the basis for security, especially when proving physical proximity. Devices could either use their actual location, i.e. the devices know where they are, or use a simpler notion of relative location, i.e. devices are at the same location although they cannot determine where this location is. Devices can also extract features from their surrounding, like ambient sound or wireless signals, to derive shared keys or proof their proximity. This process could be thought of as 'fingerprinting', where the devices look to extract unique, yet stable, features from its environment. Finally, devices can partake in a specified activity and observations made during this activity, such as device movement, can be used to derive shared keys or proof that they were taking part in the same activity and are therefore at the same location.

Communication methods involve the physical properties of the communication channel between the devices. This concept focuses specifically on practical methods to manipulate the physical aspects of the channel to achieve initial key exchange between low-resource devices. Similarly, the use of auxiliary or out-of-band (OOB) channels are discussed here for the sake of completeness but this area warrants its own survey (Mirzadeh et al., 2014). Apart from explaining the basic concept we concentrate on auxiliary channels that rely on tangible physical properties to transmit data in a 'location

Table 1 – Summary of approaches on environment context security.

Approach	Metric	Security enhancement	Benefits	Issues
Use Device Location	Device location co-ordinates	Proximity	Secure combines with cryptography	Device must know its own absolute location
Dongle Proximity	Device can hear dongle signal	Proximity/Pairing	Proximity using conventional channel	Device proximity to set beacon only
Time-of-Flight (ToF) Distance-bounding	RTT to device mapped to distance	Proximity	Between any two devices, no need to know device location	Needs special channels

limited' manner, such as vibration, rather than simply using additional conventional channels. Channel parameters, because of multipath propagation, is also a resource for key generation schemes. Both devices use common channel parameters, such as Channel Statement Information (CSI) and Received Signal Strength (RSS), as 'fingerprinting' to derive a shared key.

Device-based methods take into consideration the intrinsic properties of the devices involved. These properties are the source of unique features that can be observed within radio and audio analog front-ends of devices by using suitable 'fingerprinting' of this hardware. Similar device-dependent features can also be derived from computational processes of the devices, which can be used to identify devices or yield hardware-dependent responses to authentication challenges.

3. Environment

When making use of the physical context to assist with the security of device communication, then arguably the most obvious context type is the environment in which the communication is occurring. Within the devices' environment, there are three possible sources of physical context. The first is the location of the devices, the second is external forces acting on the device, and the third is simply ambient properties of the environment.

If the location of the devices, either their actual position or their physical position relative to each other, can be securely determined, this can obviously be used to prove device proximity. In other words, if two devices are shown to be present at the same location then it is unlikely that they are being subject to a relay attack. Location can also be used in key pairing services, although this often requires a trusted third party being present at the same location.

Devices can also be required to participate in a predetermined activity during which they can measure the external forces in effect. The measurements during the execution of the common activity can be used to derive a shared secret key. By implication, if both devices exhibit that they share a key they must have taken part in the same activity and are therefore verified to be present in the same location.

Taking measurements of their ambient environment can assist in proving that devices are present at the same location. The reasoning is that if they are observing a relatively random process linked to this location at this specific time, such as background noise or radio signals, then they are both present. Devices can also then use these measurements to establish a shared key. In Table 1, we summarize these methods which using environment context to enhance ad-hoc communication security.

3.1. *Location*

Location is an obvious way to detect a relay attack between two devices. If the devices find themselves at the same location it implies that they are in close proximity. In such a case, two different approaches to 'location' can be followed depending on the context of the communication: absolute or relative location. Absolute locations mean that the actual physical locations of both devices are known, i.e. we know where these devices are within a known coordinate system. This is feasible given that some mobile devices increasingly have an in-built GPS receiver. Relative location is the devices' position as measured against the position of the other device or against a landmark in the communication area. In other words, we can verify that the devices are at the same location but in the bigger picture we do not care where this location is, e.g. both devices are in the same room but we do not know which room or where this room is.

3.1.1. *Absolute location*

Absolute location can be used to make context-related authentication possible, even in simple devices, although these devices must have the capability to determine its own location. Hu et al. (Hu et al., 2003) proposed the notion of packet leashes to prevent relay attacks. In this scheme, all devices can determine their own location. When a device transmits a message it includes an authenticated, i.e. message is signed or subject to a message authentication code, indication of the transmitting device's location. When the message is received, the receiver can check whether the sender is at the same location. Devices need to broadcast packets to build a geographical table. A discrepancy in source location when compared to its own location would indicate a relay attack.

Location can also be used to enable a service only in a specific location. An absolute location approach has been proposed that uses a standard RFID-equipped device with a primitive GPS device. The RFID device will only respond to a communication request for information from a reading device when its current location matches that of a set of known reading locations or if the device is traveling at certain speeds (Ma et al., 2013). This is intended for use in automated tolling systems but could also be applied to general NFC-enabled payment applications if payment locations are known. The

downside to this implementation is the need to store and update the allowed reading locations, as new positions are added or old positions are revoked.

3.1.2. *Relative location*

The advantage of using relative location is that devices do not need the capability to determine their actual location so there is no need for GPS components or an overarching localization system. Relative location methods are predominantly used to prevent relay attacks, with distance-bounding being the most prominent approach. There are, however, proposals for key pairing based on relative location through the use of trusted third party beacons. The basic idea of such schemes is to equip certain locations with transmission beacons. Any devices at these locations can receive these beacon transmissions, which can then be used as the basis for setting up a shared key with other devices at this location (Studer and Perrig, 2010). The absolute location of the devices is not important, as the core requirement of the scheme is just that both devices must be in close proximity of the beacon.

Based on relative location and the beacon, it can also determine whether a device is in a particular location. Studer and Perrig (2010) proposed a protocol called User Location-specific Encryption (MULE) to prevent data leakage due to stolen laptops. In this protocol, the laptop only gives users access to sensitive files when it is in a trusted location, otherwise, it will log off or put the computer sleep. MULE utilizes specific location information from trusted locations to authenticate devices and provide security services. This protocol only gives security to sensitive files, normal files can be accessed without authentication.

Distance-bounding is a method specifically designed to cryptographically prove the relative distances between two devices. In this approach one device is proving his proximity, the prover, and the other, the verifies, is verifying the claim of the prover. It involves the timed exchange of carefully constructed cryptographic challenges and responses. The round-trip time of the cryptographic challenge-response pairs is then used to calculate the distance between the communicating parties (Hancke et al., 2009). There are three main types of attacks that are addressed by distance-bounding protocols:

1. *Relay Attack* A third party tries to convince the verifier that the prover is within close proximity by attempting to relay the challenge-response exchanges.
2. *Distance Fraud* The prover is fraudulent and is attempting to convince the verifier that it is nearby even though it is far away.
3. *Terrorist Attack* The prover attempts to co-operate with a third party, who is physically close to the verifier, in order to convince the verifier it is in close proximity. The prover shared enough information with the third party to pass a single distance-bounding protocol.

There are numerous approaches to designing distance-bounding protocols (Gildas et al., 2018; Hancke, 2012; Hermans et al., 2013; Yang et al., 2018), etc. Objectively comparing distance-bounding protocols remains a challenge. It is possible to quantify the success probability of each of the three given attacks for each protocol for a given number of rounds, but additional aspects such as execution time or memory consumption of additional rounds should ideally also be considered.

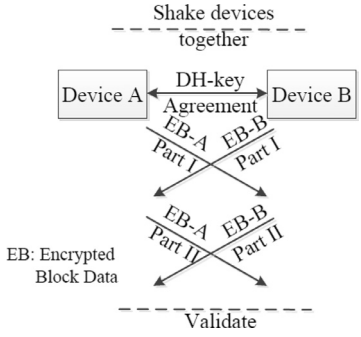

Fig. 4 – ShaVe sequence diagram.

A framework has been defined that attempts to allow for a standardized approach to cryptanalysis and designing new distance-bounding protocols (Avoine et al., 2011). This formal framework provided a consistent attack model for evaluating protocols.

The main drawback to distance-bounding protocols is that the timed exchanges need special channels to achieve accurate round-trip time and to maintain the security of the overall protocol (Hancke and Kuhn, 2008). This means that distance-bounding techniques have not seen wide adoption within deployed systems. There are numerous channels that have been implemented to support distance-bounding protocols (Hancke, 2011; Ranganathan et al., 2012; Rasmussen and Capkun, 2010). However, these are only prototype channels under controlled conditions and more work is needed to construct practical channels that are both feasible in actual devices and suitable for distance-bounding.

3.2. *Common activity*

Common activity refers to a set of physical actions that can be performed by both devices. Physical context observations that these devices collect during the activity could be used to generate a shared key or help verify that devices are in the same location.

A simple common activity is to take two devices and move them together. During the movement, both devices monitor accelerometer sensor data. This sensor data is then used as a means to perform proximity authentication, i.e. both devices were subjected to the same external movement, and key exchange, i.e. since both devices have made similar measurements these could be used to derive a common key by each device (Mayrhofer, 2007). 'ShaCK' (Mayrhofer and Gellersen, 2009) is a similar method using accelerometer sensor data to generate a shared key, but it continually needs to exchange parts of the collected data each to evaluate the similarity. Only if the data is within a set threshold for similarity does this method collect the measurements for key derivation.

"ShaVe", another method proposed by (Mayrhofer and Gellersen, 2009), is a method using a motion channel to generate the shared key through the accelerometer data generated by shaking devices together, as illustrated in Fig. 4. First, two

devices generate shared keys using standard Diffie-Helman (DH) key agreement via insecure channel. Then they exchange encrypted block of acceleration data via an Interlock protocol (Rivest and Shamir, 1984). This protocol is used against man-in-the-middle attacks. As the data is encrypted by block ciphers, this protocol split each encrypted message block into several parts. Devices are required to transmit the first part and then never transmit the next part until receiving one part of the message block belonging to the other device. After this procedure, the device compares the received data with its locally captured data with a frequency domain coherence measure. If the check result is successful, these two devices can communicate with each other with the agreed key. But the security level of this method is the same as the standard DH key agreement. The comparison of acceleration data does not add any security if the attacker can get the DH key through an MITM attack. The attacker can wait for the completion of the second and third step and then communicates with both devices using the key.

An activity that can be performed is the moving of both devices to enable protection against message sender identification/location. The approach considers that the system for key exchange can be built upon an approach already created, where the focus is on protecting the identity of the sender rather than the contents of the message (Castelluccia and Mutaf, 2005), as shown in Fig. 5. During the preparation period, device A will send a start signal containing the length of the key to device B and also the address of A. B will send a start signal with the address of B back. Then during the key generating period, they begin to send packets randomly. Each packet also stores an address from A or B randomly as a source address and the other as a destination address. If the source address of a packet is right, the bit at the corresponding (which is corresponding to the sequence of this packet) position of the final key would be 1, otherwise it would be 0. To make this key agreement secure, we shake both the devices together during the exchange to achieve spatial indistinguishability. In other words, during the key exchange procedure, the user should shake the devices to keep changing their position relative to each other. Thus, attackers cannot distinguish the sources of messages, e.g. A is on the left and B on the right from directional signals, and build the secret key. Finally, A and B will exchange hash values of addresses of A and B and secret key. If both are the same, it means the key exchange is a success. This method can deal with a majority of analysis attacks based on directional RSSI-signals.

3.3. *Fingerprinting in environment*

Environment-based fingerprinting schemes usually extract features from ambient sources, such as background audio and radio signals and radio. These ambient sources are random processes over time so to obtain similar measurements devices must take measurements in the same location during the same time period. These schemes are mostly intended to establish shared keys. However, there is also an implicit proximity proof because the devices end up with the same key based on extracting features from their environment. This implies that both devices had to be present in the same environment during the feature extraction.

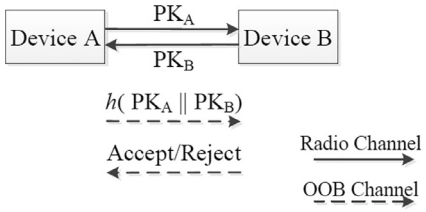

Fig. 5 – Explicit authentication method.

Schurmann and Sigg (2013) extracted an audio fingerprint from ambient audio. This paper extracted the key with energy difference in the frequency domain, which is robust to processing and combined fuzzy commitment scheme with Reed-Solomon error correcting code to make sure two devices can exchange messages securely with the fingerprint got by themselves. The experiments show that the Hamming distance between fingerprints extracted from identical ambient audio is much lower than those extracted from nonidentical ambient audio. It means that with a well-chosen configuration of the Reed-Solomon error correcting code, only devices with identical ambient audio can communicate with each other successively.

Quach et al. (2014) extracted an audio fingerprint from background voice. Because of the little difference between audio signals recorded by two devices, this paper extracted the key with energy difference in the frequency domain, which is robust to the processing and combined fuzzy commitment scheme with the Reed-Solomon error correcting code to make sure two devices can exchange messages securely with the fingerprint got by themselves. The experiment showed it worked well between two phones of the same type, but it did not consider phones of different types.

4. Communication

The next area that can be used to generate an additional physical context for security services is the communication channel. Ad-hoc communication between embedded devices is most often across a wireless channel. In this section, we focus on methods that use tangible properties of the wireless communication for key pairing and relay resistant communication security for embedded/mobile devices. The three approaches we look at are friendly jamming, auxiliary channels and channel parameters. The topic of auxiliary channels is a subset of a larger area of research on out-of-band (OOB) communication, i.e. using multiple channels, for security. For example, if we make an online purchase via a wireless network we might get confirmation of a payment via mobile text message. OOB is a promising direction, especially for device pairing, for security services in ad-hoc communication, so we include it in our discussion. However, we focus on OOB channels that exhibit a tangible physical property that differs from conventional wireless channels, such as audio, visual or motion-based communication. In Table 2, we summarize these methods that use communication context to enhance ad-hoc communication security.

Table 2 – Summary of approaches on communication context security.

Approach	Metric	Security Enhancement	Benefits	Issues
Friendly Jamming	Wyner's wiretape channel model	Pairing	Suited to source with constrained devices	Prone to eavesdropping
Auxiliary channel	Audio, motion and visual channel	Proximity/Pairing	Against MITM attack	Note suited to computation ability constrained devices
Channel estimation	Multipath influence of channel	Pairing/proximate	Widely implementation in practical Scenarios	Limited randomness in static environment

4.1. *Friendly jamming*

There are several approaches to using the channel characteristics to securely send data, which have their roots in the notion of the 'wire-tap' model proposed by Wyner (1975) in 1975. In this model, the sender sends data S which will be interfered by noise N_1 and N_2 during the transmission on the wireless communication channel, which affects the signals received by an intended receiver and eavesdropper respectively. The signal received by the intended receiver is interfered by N_1 as $S_r = S + N_1$ while the signal received by an eavesdropper is $S_e = S + N_2$. If N_1 is much less than N_2, then the intended receiver can recover the original data but the eavesdropper cannot. To make sure that N_2 is much larger than N_1, it is proposed that systems introduce an artificial noise signal to jam the communication channel – co-operative or 'friendly' jamming. This makes the approach more practical and can provide some guarantee that N_1 is much less than N_2. This is in contrast to the general concept of jamming, which is intended as a denial of a wireless service.

The secrecy capacity, meaning the highest reliable transmitting rate of secured data over a channel, is used to assess the security bounds of such channels proposed for physical layer security. Secrecy capacity means that there exists a capacity under which the eavesdropper can get almost no useful data. The highest secrecy capacity is often considered as the capacity difference between the good channel (from the transmitter to the receiver) and the bad channel (from the transmitter to the eavesdropper). Many papers have described research on bounds of secrecy capacity in different type of channels: Gaussian wiretap channel (Tang et al., 2008; Tang et al., 2011), Rayleigh channel (Li et al., 2007Lai and El Gamal, 2008), Gaussian multiple-access wiretap channel and Gaussian two-way wiretap channel (Tekin and Yener, 2008), fading channels (Gopala et al., 2008) and non-general channels (Vilela et al., 2011; Zhou et al., 2012). But these methods only analyzed the secrecy capacity based on an attack assumption that there existed only one eavesdropper or multi-eavesdroppers without collusion.

In this section, we focus on friendly jamming approaches proposed for low-resource, ad-hoc communication. The general principle is that the data and the noise will be transmitted at the same time and that the noise signal is known to the receiver. The receiver can then cancel the noise from the combined noise-data signal, but an adversary cannot, which allows the channel to securely send data. These approaches are divided into two categories, illustrated in Fig. 6, depending on which party is responsible for the jamming signal. The jamming signal can either be generated by an additional device, e.g. two RFID tags respond to a reader request with one sending data and the other noise, or by one of the communicating devices, e.g. the receiver sends noise to cover data sent by the sender.

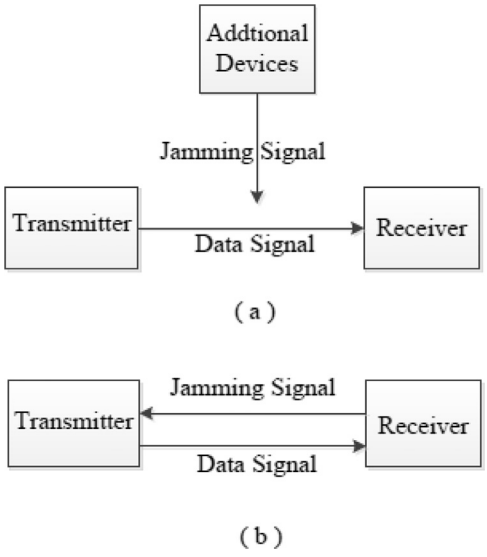

Fig. 6 – Friendly-jamming approaches: (a) Additional device(s), (b) Self-jamming.

4.1.1. *Jamming by Additional Devices*

In this category, additional devices are expected to transmit jamming signals. To cancel the jamming signal and recover the data, however, the receiver must be able to determine which signal was sent by the jammer. The additional device and the receiver must, therefore, have a pre-shared key, or another prior relationship, such as a code-book (Tang et al., 2011), to generate the same noise signal or the same codebook to decode the jamming signal to recover the data signal. But it is different when the receiver and device share the same codebook.

Castelluccia and Avoine (2006) proposed a bit-blocking protocol to securely transmit secret data in the context of an RFID system, as shown in Fig. 7. The reader and the noisy tag share a key K. When the tag transmits data to the reader, the noisy tag will also transmit data. The reader first broadcasts a nonce N which is generated randomly. Then noise node will generate

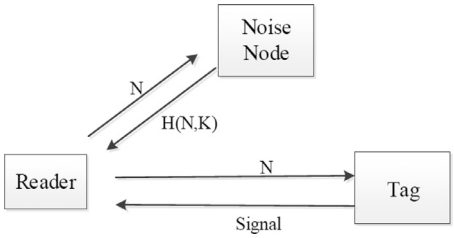

Fig. 7 – Infrastructure and procedures of bit-blocking protocol.

a sequence of bits by random function *H* with *N* and *K*. Next, the tag and node will send data to the reader simultaneously in bit level, which means they will send a bit simultaneously at a time slot. Except for the noisy node, only the reader has the knowledge of *H*, *N* and *K* to recover the data sent by the tag. But there is a problem when the node and tag send the same bit, the information of this bit will be leaked. The authors suggested using a code-based protocol instead of bit-based protocol. A code-based protocol does not transmit every bit of the data directly but converts every bit into a code to transmit.

Shen et al. (2013) also proposed a method that needs the receiver and jamming node to share the same key, but it aims at a multiple jamming nodes environment. In this environment, all nodes must be synchronized in order to correctly cancel the jamming signal. Each jamming node is allocated a certain frequency called a pilot frequency. First, these jamming nodes generate random jamming signals by the shared key. Then a noise signal of pilot frequencies is added to the random data signal generated by the same jamming node separately. These pilot frequencies have a quite larger magnitude in the frequency table than the other frequencies. The receiver can distinguish signals from which jammers by these pilot frequencies.

4.1.2. *Self-jamming*

In this category, the jamming noise is transmitted by the receiver while the sender transmits data, as shown in Fig. 6. The noise cancellation is simpler than in the case of a third-party jammer, as the receiver already knows the noise signal itself transmitted.

An approach that can achieve self-jamming without shared information has been proposed in RFID systems (Fei et al., 2013). RFID systems have some inherent properties that fit well with this approach: the reader is transmitting a radio carrier to the tag at all times to power the tag, and the tag uses back-scatter or load modulation to transmit data, i.e. the tag modulates the carrier of the reader and not a carrier it transmits itself. This noisy reader can, therefore, transmit a noisy carrier signal while receiving the sent signal from the transmitting tag, as the tag can just modulate the noisy signal. The reader, who is the device adding the noise, is then able to remove this noise in order to retrieve the message that was sent. This approach is considered a practical method for mitigating against passive eavesdropping. The authors made the reader signal increasing with a constantly increasing value by the same interval. In overview, the reader signal increases from minimum amplitude to maximum amplitude periodically. The jamming theory is shown in Fig. 8. It shows that the eavesdropper can not distinguish the bits with sequence 10 as the amplitudes of these two bits are the same. But according to another paper (Hu et al., 2015), this method is vulnerable due to the unsynchronization of the reader signal and the tag signal. So this paper improved the vulnerable method by randomizing the interval and the increasing value.

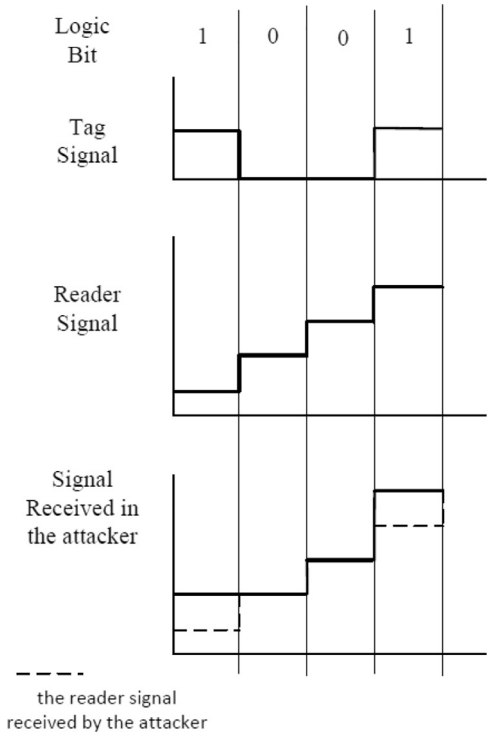

Fig. 8 – Reader signal assisted jamming method.

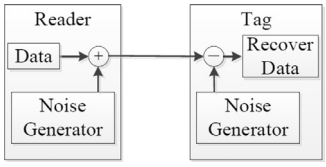

Fig. 9 – RFID noisy reader.

A similar approach has been proposed in the context of RFID devices (Savry et al., 2007). This approach requires that both reader and tag have the same pseudo-random noise generator and the same start value, which means these two noise generators can generate the same noise. Details are shown in Fig. 9. The reader adds noise to data and the tag cancels noise with the same noise generator. This principle has been demonstrated for ISO/IEC 15963 RFID systems (Achard and Savry, 2012).

A different kind of approach is proposed to achieve key agreement (Jin et al., 2014). It looks similar to bit-blocking (Castelluccia and Avoine, 2006) but it has a specified coding way, which is illustrated in Fig. 10. This varied bit-blocking is vulnerable to unsynchronization. The first valley and peak in Fig. 10 can deduce that bit 0 or 1 is coming from which device. To overcome this problem, the authors change the start time

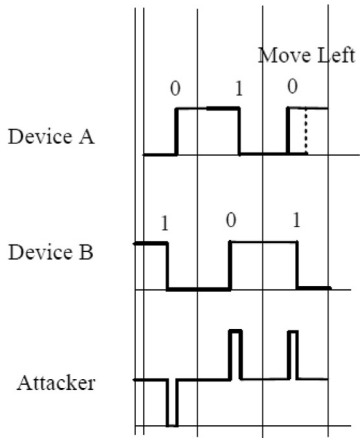

Fig. 10 – Practical key generation by jamming.

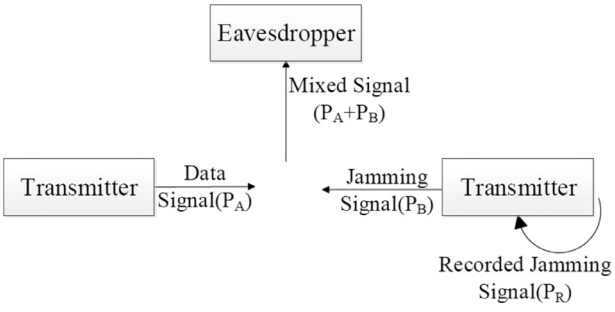

Fig. 11 – Dhwani method for self-jamming audio communication.

of the bit a little earlier or later. Fig. 10 shows that the move of the start time can cause the peak no matter bit 0 coming from which device.

This approach has also been proposed to secure non-radio communication between mobile devices. A secure acoustic communication method was proposed in by Nandakumar et al. (2013), as shown in Fig. 11, that uses sounds and the 'wire-tap' model approach to send data. The approach is that the receiver can only recover the data a set sign-to-noise-ration (SNR) above a set threshold SNR_{min}. The goal is therefore to decrease the SNR of the eavesdropping but maintain an SNR above this threshold for the receiver. Fig. 11 shows that when device A and B send a signal with power P_A and P_B separately, the eavedropper will receive a mixed signal with power $P_A + P_B$. The power of noise P_B is larger than the power of data P_A, which leads to the unrecoverability of the data. But the condition is the same for device B. In order to make device B recover the data, device B recorded the noise in advance with power P_R. So we can remove the noise in the mixed data, which leads to a SNR higher than SNR_{min}. They evaluate the security of their scheme based on an assumption that there is only one passive eavesdropper.

This type of scheme has been demonstrated to work well for the prevention of passive eavesdropping attacks. One of the challenging aspects of this type of research is estimating the secrecy of these channels. Different authors use different attack models and even though schemes are shown to be secure there is now a common framework for security analysis for what attack model is most applicable. The question should be asked if a single, passive attacker is the only valid attack model? According to Pinto et al. (2009), colluding eavesdropping can dramatically decrease the secrecy capacity with the increasing amount of eavesdroppers. (Tippenhauer et al., 2013) also showed that if the distance between eavesdroppers is less than that between the transmitter and the receiver, using radio communication, eavesdroppers can recover the data secured with friendly jamming. PriWhisper (Zhang et al., 2013) makes a similar observation for audio channels, implementing the same approach as Dhwani, by showing that the scheme is resistant against multiple colluding eavesdroppers' attacks on condition that the two devices are less than 3cm from each other. Security also depends on the nature of the jamming noise and the data sent. But as this method needs receivers to transmit noise with the highest volume during all of the data transmission periods, users may feel annoyed.

4.2. *Auxiliary channels*

Auxiliary channels include methods making use of additional channels for communication. These methods are designed to use auxiliary channels instead of common radio channels to complete part or the whole cryptographic protocol. Researchers believe auxiliary channels are more secure than the traditional insecure wireless radio channel and some researchers also introduce the human device owner as a trusted third party to validate the shared key through his actions. These channels can be divided into four groups: light, sound, motion and other communication channels. Motion channel includes vibration and button pressing. Other communication channel includes infrared, physical 'touch' and additional radio technology. Auxiliary channels can deal with relay attacks (Stajano et al., 2010) according to transmitting secure data via an unrelayable channel (can be achieved by auxiliary channels).

4.2.1. *Audio channel*

Halperin et al. (2008) proposed a method with the help of an audio channel to protect the security of Implantable Medical devices (IMD) as IMD is in a human body and always computationally constrained. They added a passive tag to the IMD. The procedure is as follows: First, a reader emits an unmodulated radio frequency signal to energize the passive tag to let it calculate a key randomly. Next, IMD will send the key using audio wave by a microphone. This method needs the reader and the IMD close enough to each other and it believes that in such a way the eavesdropper cannot recover the data. This method is a simple idea, but very practical and harmless to humans, but it is prone to eavesdropping (Halevi and Saxena, 2010).

A method using an audio channel to exchange both messages and authentication information is known as HAPADEP (Human Assisted Pure Audio Device Pairing) (Soriente et al., 2008). It just uses an audio channel to exchange their keys. Also, it utilizes the user (human component) as the counter to an MITM attack. Operations are shown in Fig. 12. First, two devices will encrypt their keys and transform them to audio sequence. Then one device will play the audio sequence

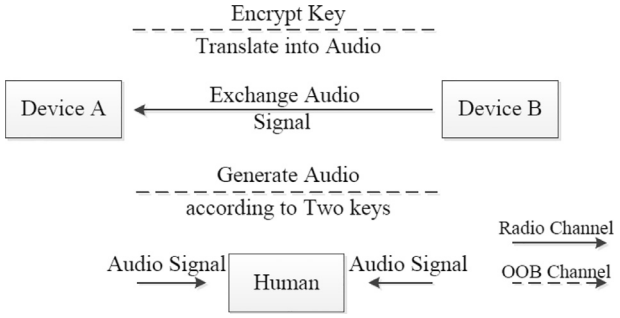

Fig. 12 – HAPADEP sequence diagram.

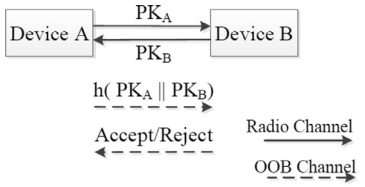

Fig. 13 – Visual integrity checking protocol.

and then the other. After they decode the other device's key, they will add these two keys together in the same sequence and also transform keys into an audio sequence. Finally, these two devices will play audio together for humans to compare. 'Loud-and-Clear' (Goodrich et al., 2006) also has a similar mechanism but it replaces one device with a visual channel. After the key has exchanged, one device transforms keys into text and the other transforms keys into audio heard like someone is reading the text. These methods are secure enough, but they need more energy than those devices that only use asymmetric cryptography, which is not suitable for computationally constrained devices.

4.2.2. *Visual channel*

McCune et al. (2005) implemented a method named as 'Seeing-is-Believing'. They chose a visual channel as an OOB channel and transformed the hash value of public keys into a bar code that enabled the other devices using a camera to read the messages. This method has a limitation that both devices should have cameras and screens. Saxena et al. (2006) presented this protocol to overcome the limitation in Seeing-is-Believing. He solved this problem by using this protocol that only needs a one-way visual channel. The scheme process is shown in Fig. 13. Device A and device B exchange a public key at first, then device A sends hash values of public keys of A and B to device B via a visual channel. At last, B shows the result to the user. The security of this protocol depends on the hash function and the length of public key. However, it is potentially prone to an MITM attack as a human cannot compare final keys of A and B, although this would involve a visual relay channel.

A method involving the use of a visible light channel to facilitate secure key deployment within an application has been presented in Perkovic et al. (2012). This method involves a novel multi-channel group message authentication protocol, in which information is transmitted over both a radio and a visible light channel enabling the secure deployment of an application key. One of the main issues found when attempting to secure devices is the lack of quality output interfaces and the corresponding receivers (this is especially true for embedded applications). A security scheme that is universally applicable is ideal in such a situation. Such a scheme has been developed and has been applied across a range of applications (Prasad and Saxena, 2008). The scheme is based on the comparison of short and simple transmissions involving synchronized patterns. This again requires a manual human interaction within the system. This scheme has been improved upon and automated by making use of the visual auxiliary channel, commonly available on a device through hardware such as a digital camera (Saxena et al., 2008). It has been shown that the automated approach is more users friendly and generally faster than the non-automated manual scheme. Importantly, it is also shown that the proposed scheme is accurate in the detection of any possible attacks and allows for appropriate steps to be taken once an attack is identified.

4.2.3. *Motion channel*

Saxena et al. (2011b) presented a key exchange approach using a motion channel. In this scheme a human user pairs the sender, e.g. a mobile device, with an RFID tag. The mobile will generates a key and transmit the key through controller vibration. The user should keep mobile touching the tag during the vibration and the tag must be equipped with an accelerometer. The accelerometer in the tag will sense the vibration data collected for the tag to restore the key and complete the authentication. The vibration encoding method is as follows: The key to be transmitted is a binary sequence starting with three additional bits "110". If the phone keeps vibrating for 200ms it means bit "1" and no vibration for 200ms means bit "0". After the end of the key transmission period, the two devices can securely communicate with each other using the shared key. This method is simple and effective, but it is prone to the eavesdropping (Halevi and Saxena, 2010) if the resultant audio signal of the vibrating devices can be recorded.

Soriente et al. (2007) proposed a method using button pressing as auxiliary channel. This method just requires a button and display as indispensable equipment. First, one device exhibits fully random passwords that span the full n-digit vector space. Then, the user tries to press the buttons of two devices simultaneously according to the password. After the user finishes this action, two devices are paired. Security is obviously based on the password transmitted by button pressing. It is also prone to the eavesdropping (Halevi and Saxena, 2010) based on listening to the button presses.

Kriara et al. (2013) also used this basic method to identify RFID devices. Here a reader can pair passive RFID tags with each other by recognizing the movement (or gestures) of tags in its reading range. If the reader believes that tags have been performed the same movement (or gestures), then the reader considers both tags to be present and will facilitate the pairing of these tags. The experiment shows high accuracy in recognizing circle motion and over 18% mismatch rate in line motion.

Chagnaadorj and Tanaka (2013) use accelerometer data to identify devices taking part in communication. The procedure is shown in red (Fig. 14). First, two devices exchange data and

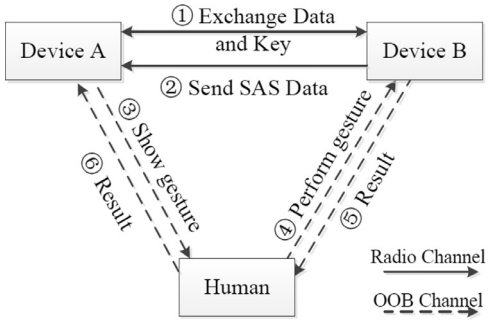

Fig. 14 – Procedures of mimic gesture.

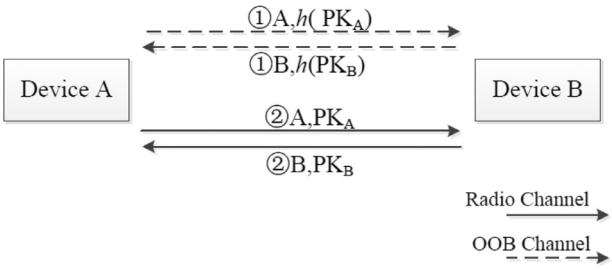

Fig. 15 – Talking to strangers.

a public key used for Short Authentication String (SAS) protocol. The SAS protocol is proposed by (Vaudenay, 2005) to authenticate the communication in a normal insecure wireless channel with the help of an OOB channel by exchanging short strings via an OOB channel. Device B then generates some gestures, converts them to SAS data and sends this data to device A. Device A converts the received data into gestures and shows them to the user in sequence. The user holds device B and performs gestures one by one in sequence. Next, the accelerometer data recorded by B will be analyzed and restored to gestures. Device B will compare these gestures with previous gestures to validate device A and give the comparison result to device A via the user.

4.2.4. *Other communication channel*

Stajano and Anderson (2000) proposed this method to securely bootstrap in ad hoc wireless networks. They imitated the behavior of duckling in their method. This method chooses the first moving object that makes a voice like the mother of a duckling. This is the first step of the trust establishment procedure. Then each other object, the duckling, will take the first device that sends it a secret key as its owner. To protect the confidentiality and integrity, they used standardized physical interfaces and cables that belong to other non-radio channels.

Balfanz et al. (2002) presented a method to authenticate the key establishment in ad-hoc wireless network. It compared work similar to the paper mentioned above. It also enhanced the Resurrecting Duckling model by using infrared communication as the OOB channel instead of physical connect and needed a bit help of users. This method is shown in Fig. 15.

First, the two devices initially send their addresses with the h(PK), a one-way hash function and the argument of this function is the public key, to each other via the infrared or audio channel. Then, they just exchange their public keys using the common insecure wireless channel. Finally, they compare the messages they receive in two steps to validate the public keys.

4.3. *Channel parameters*

Channel parameter is the source of random channel characteristics. Taking advantage of the randomness of the channels, two wireless devices can establish a security relationship. Temporal variation, channel reciprocity and spatial decorrelation provide support for the implementation of key pairing schemes.

Temporal variation means that as the receiver, transmitter or any objects in the environment moves, the reflection, refraction and scattering of the channel paths also change. These changes provide resources for key generation.

Channel reciprocity implies that the multipath properties of the radio channel (gains, phase shifts, and delays) at two devices connected by the same link in time and on any given frequency channel are identical. It makes sure that two devices can extract the same key.

Spatial de-correlation is a principle used to protect legitimate devices from an attacker. An attacker will experience de-correlated multipath fading if he is more than one half-wavelength away from the legitimate device and cannot recover the secret key. This principle achieves location distinction that aims to detect devices' location changes, movement or facilitate location-based authentication (Fang et al., 2017).

By using these three principles, there are two different channel parameters for key extraction between two devices, which are received signal strength (RSS) and channel statement information (CSI). CSI is a fine-grained channel parameter and RSS is coarse-grained.

RSS refers to a power level of a radio signal received by the receiver. The higher the RSS number, the stronger the signal. Although longer propagation distance leads to lower RSS, RSS does not decrease linearly as the distance increases. RSS is affected by many other factors, including the location of the antenna, the antenna itself, the number of obstructions in the proximity of the devices, and the environment. So a set of transmitter and receiver will obtain a special RSS, which can use to generate a shared secret key.

CSI describes the combined effect of a multipath channel when a signal propagates from transmitter to the receiver. This method is also called channel estimation and channel response. A signal transmits through different paths will experience different scattering, fading, reflection and refraction, which makes the combined effect different and guarantees a high Key Generation Rate (KGR). In this paper, CSI divides into channel impulse response (CIR) and channel frequency response (CFR) and we will illustrate these respectively.

4.3.1. *RSS*

The transmitted signal $x(t)$ experiences a multipath channel and the received signal $y(t)$ can be given as

$$y(t) = \int_0^{\tau_{max}} h(t)x(t-\tau)d\tau + n(t) \quad (1)$$

where $n(t)$ is the noise in the channel, τ_{max} is the maximum channel delay, $h(t)$ is the channel response of the

transmitted signal. RSS is used widely in practical implementations, as most of the current off-the-shelf wireless cards can measure it without any modification. It has been prototyped in not only wireless local area networks (WLANs), but also wireless sensor network (WSNs).

A WLAN is a network that links several devices using wireless communications within a limited area. Jana et al. (2009) evaluated RSS-based key generation in real environments under real settings, in both static and dynamic environments. They found that the entropy of RSS measurements in a static environment is very low and an adversary can cause predictable key generation. Based on this issue, they develop an environment adaptive secret key generation scheme that uses an adaptive lossy quantizer in conjunction with Cascade-based information reconciliation and privacy amplification.

(Premnath et al., 2013) expanded on Jana et al. (2009). They evaluated the performance of secret key generation using small, low-power and hand-held device. They also used Multi-Input Multi-Output output (MIMO)-like sensor networks and tried to achieve a high KGR. High bit mismatch caused by multiple sensors can be solved by an iterative distillation stage.

Guillaume et al. (2015) also realized a physical-based key generation in practical environment and compared them. They presented results for the KGR and achieved key quality.

WLANs have become popular to use at home, where there is limited mobility and the key generation scheme is vulnerable to an adversary. The KGR is very low in such static environment. To address this issue, researchers use multi-antenna (Zeng et al., 2010) or adaptive channel probing (Wei et al., 2013), and they try to decrease error bit rate in key extraction by using a level crossing algorithm (Mathur et al., 2008).

On the other hand, in WSNs, the communicated entities are sensor nodes. In order to address the limited KGR under a static environment in WSNs, randomness in the frequency domain is exploited (Wilhelm et al., 2013). Body area networks are wireless networks of wearable computing devices, and is a special category of WSNs. There is an implementation in this area (Ali et al., 2014). RSS-based key generation systems also can be used in vehicular communication (Zhu et al., 2013).

Liu et al. (2012) proposed to generate group shared keys among multiple devices leveraging RSS. They also developed two protocols ensure security called star-based and chain-based.

4.3.2. CSI

CIR

As a signal propagates through a multipath channel, more than one signal will be received at the receiver and the wireless channel state $h(t)$ can be described as

$$h(t) = \sum_{n=0}^{N-1} \rho_n(t) e^{j\phi_n(t)} \delta(t - \tau_n(t)) \tag{2}$$

where $\rho_n(t)e^{j\phi_n(t)}$ is a complex number, which represents amplitude attenuation and phase shift of the composite channel. N is the number of paths, $\tau_n(t)$ is the time delay of the $n^{(}th)$ path. These impulse responses are time-varying in presence of time variation of the geometrical reflection and reflection conditions, so amplitude and phase information can be used for key generation.

The channel estimation consists of amplitude and phase components, both of them are able to derive a key separately as they vary independently. (Wilson et al., 2007) and (Marino et al., 2014) are amplitude-based key generation. But the amplitude is vulnerable for an active adversary as a powerful attacker can manipulate the amplitude of the signal received by legal devices. The amplitude is a multiplicative component. By increasing or decreasing the amplitude of the transmitted signal, an adversary can change the signal's amplitude dramatically and influence key extraction. In contrast, phase information is an additive component and the range of phase is from 0 to 2π. Even if the adversary can add an arbitrary phase to the received signal, the legitimate devices compute their differential phase, the adversary has no control over the change in the channel's phase during the same interval. Differential phase secret bit extraction can tolerate a powerful attacker who controls the transmission source. Therefore, although many researchers have focussed on amplitude-based key generation, there is only one practical implementation in phase-based key generation, called ProxiMate (Mathur et al., 2011).

In ProxiMate, two devices in close proximity can perceive the same small-scale fading and derive a shared key. In contrast, an adversary who is not close to the legitimate device will experience different small-scale fading as it is half of the wave away from the legitimate device and the adversary cannot derive the same shared key. So ProxiMate can be used to prove proximity and generate a shared key.

However, an adversary can mimic devices' CIR by manipulating multipath channel characteristics. The adversary can transmit different versions (different amplitude and time delay) of the original signal and superimpose them together to mimic real multipath channel characteristics, then remove their own multipath response by reverse-engineering existing wireless channel estimation algorithms and performing linear transformations on the original signal (Fang et al., 2017). Fang et al. (2017) proposed an auxiliary receiver to defend against this attack.

CFR

CFR is a parameter of the frequency domain and can be described as

$$H(f) = \int_0^{\tau_{max}} h(t) e^{-j2\pi ft} dt \tag{3}$$

where τ_{max} is the maximum channel delay.

CFR works for Orthogonal Frequency-Division Multiplexing (OFDM) systems. OFDM is used widely in wireless communication systems, such as WLAN, WiMAX and 3G LTE. It can provide higher data rate by exploiting both space and frequency diversity. In OFDM systems, the bandwidth divides into multiple subcarriers and each subcarrier can be regarded as a narrowband channel, which can provide more CSI and higher KGR.

As opposed to CIR, phase estimation is not used in CFR as it is impacted by the time and frequency offset. Liu et al. (2013), Xi et al. (2014) and Zhang et al. (2015) are the practical implementation of amplitude-based key extraction.

Liu et al. (2013) provided higher bit generation at 60–80 bit/packet. However, it cannot keep low bit mismatch rate because of non-reciprocity of two devices due to devices' different disparity of electrical characteristics. To deal with this

Table 3 – Summary of approaches on device context security.

Approach	Metric	Security enhancement	Benefits	Issues
Device fingerprinting	Transceiver's difference	Proximity	High accuracy	Think about feature stability and affected by environment
Internal hardware	Physically unclonable function	Proximity	Efficient and robust	Need to consider attacks

problem, we need information reconciliation. In this paper, they proposed a novel channel gain complement (CGC) algorithm.

Another problem for key generation in OFDM systems is that CSI measurements from adjacent subcarriers may have strong correlations, which is vulnerable to key cracking attacks. To address this issue, Xi et al. (2014) proposed a fast secret key extraction protocol called KEEP. KEEP uses the validation-recombination mechanism to extract keys using the combined information of all subcarriers. It proposed a mismatch federated filtration method to reduce the bit mismatch rate.

Both Liu et al. (2013) and Xi et al. (2014) use all subcarriers to generate a shared key, Zhang et al. (2015) proposed to use individual subcarriers in OFDM systems. They proved that CFR of individual OFDM subcarriers is usually a wide sense stationary (WSS) random process, which is able to find the optimal probing period and maximize the KGR.

However, in WSNs, the sensor nodes are resource-constrained and usually static or with little movement. To generate secret keys in such case, (Zenger et al., 2014) is proposed.

5. Device

The third source of physical context information is the device itself. Even though devices appear to be the same, they might be manufactured in the batch using the exact same design, they are actually small differences and unique traits exhibited by each device. This stems from the fact that the components used to make these devices. For example, a resistor or capacitors have a small degree of variability in their true values, while crystal oscillators show a small offset from their advertised frequency while also drifting at different rates. It is, therefore, an ongoing research question whether we could use such differences to distinguish devices. Such a link to the physical properties of a device could be used to prevent relay attacks as the recipient can actually verify which physical device sent the data, i.e. whether it is a proxy or the legitimate device. Research on using the physical context of the device can be classified into two categories: device interfaces and devices internals. In Table 3, we summarize these methods using environmental context to enhance ad-hoc communication security.

5.1. *Device interface fingerprinting*

Physical component based fingerprinting techniques can be used to distinguish characteristics of different devices, i.e. different models of phone, but it has also been found that identical devices, i.e. the same model device, can also be distinguished. This is due to the manufacturing variability of the electronics components and it is unavoidable in real life. Given the number of components present in a circuit, these combined minor differences can yield a unique circuit behaviour. The two electronic interfaces to devices that have been studied for fingerprinting are radio transceivers and audio speakers and microphones.

5.1.1. *Fingerprinting based on radio signal*

A radio transceiver often consists of a transmitter and a receiver that share a common circuit design. However, because of the manufacturing and component variability, the signal transmitted by the transceiver can be different for each device, even if the devices are of the same design.

Bertoncini et al. (2012) proposed a method using unintended modulations of the emitters of radio frequency tags as unique identifications. This paper compared three kinds of feature extraction methods: dynamic wavelet fingerprint (DWFP), wavelet packet decomposition (WPD) and higher-order statistics. It also compared four main types of classification methods to choose the best one. The experiments showed that it distinguishes between two RFIDs that are similar with higher than 99% accuracy. But this paper only can determine that whether device A is the same as device B, it cannot tell us the identification of device A.

Barbeau and Kranakis (Hall et al., 2006) proposed a way that can tell us whether this device is device A. This paper used radio frequency fingerprinting to extract a fingerprint from turn-on transient portion of a signal. It designed a statistical classifier and a decision filter to get the correct results by comparing signals. The experiments showed this approach worked well but it needed to increase its accuracy. (Klein et al., 2009) also did research on wavelet fingerprinting, but they used non-transient preamble signals instead of transient signals. To achieve the robust, they extracted dual-tree complex wavelet transform features as the fingerprint. They also combined Fisher-based multiple discriminant analysis with maximum likelihood classification to validate the effectiveness of wavelet fingerprinting. This work has higher accuracy than any other methods when SNRs are below 20 dB.

Danev et al. (2009) proposed an approach using features extracted from the data signal to identify RFID devices. There are two kinds of features: modulation-shape features and spectral features. For modulation-shape, the fingerprint is the modulated signal envelope. Spectral features are extracted from frequency sweep and data burst. But sweep and burst contain too many data points. This condition means noisy points cannot be ignored. To eliminate noisy points, the authors used

modified Principal Component Analysis (PCA) technology. This approach requires that devices emitting data signal should be stationary or at a fixed position with respect to the data signal receivers. Also, the emitter and receiver should be within close proximity to each other. Ureten and Serinken (2007) proposed a similar method extracting the amplitude envelope of the turn-on transient of the radio signal as a fingerprint. Bonne Rasmussen and Capkun (2007) extracted length, amplitude and discrete wavelet transform of the transient signal as a fingerprint in sensor networks.

Periaswamy et al. (2011) used minimum power response as the fingerprints of devices. This method is aimed at passive RFID tags. Passive RFID tags need energy from outside to activate them and then transmit responses. Due to manufacturing errors, the energy needed by tags that are the same model and made at the same time are different from each other. The authors recorded minimum powers that were needed to activate the passive tag to have a response at multiple different frequencies. This set of values is the fingerprint of a tag. This fingerprint is reliable and the passive tag also does not need to be modified. But when it is applied in reality, passive tags can only be identified correctly when they are at the same distance as they were in the previous experiment.

Zanetti et al. (2010) utilized a new features to identify devices. These are called time domain features, which include the time interval error (TIE) and the average baseband power. The TIE refers to the offset between real clock active edge and ideal clock active edge. Average baseband power is the mean power value of a received RN16 preamble signal. The authors also tested the stability of the fingerprint in different conditions and concluded that the TIE feature is the most stable one.

Detecting and analyzing the transient signals needs some costly and precise equipment that not widely used in common devices. Brik et al. (2008) et al. proposed a method using wireless signals formed in the modulation step to identify the transmitters of devices. During the modulation step, there are five kinds of error: phase error, magnitude error, I/Q origin offset, frequency error and SYNC correlation error. They combined them all together to identify the devices more effectively.

A method identifying UHF RFID devices by time was proposed by Periaswamy et al. (2010). When RFID tags receiving an acknowledge packet for its previous response to the reader's query command, tags will send the EPC, Protocol control (PC) and Cyclic redundancy check (CRC) to the reader. Time cost by the reader to receive these data is used as the fingerprinting of RFID tags.

5.1.2. *Fingerprinting based on audio signal*

Methods in this category often utilize manufacturers imperfections in the audio components, such as speakers and microphones, to identify devices.

Zhou et al. (2014) proposed a method extracting the frequency response features of the speaker as fingerprint to identify mobile devices. To achieve stealthy and unique fingerprint generation, the authors chose high-frequency audio emitted by the speaker to extract the fingerprints. High-frequency audio is silent to human ears and the variation between audios are much higher than other frequencies. To achieve robust fingerprint features, the speaker should play audio in a specified pattern to avoid the interference by noise as much as possible. The experiments in this article were conducted in a strictly constrained environment - a quiet office over a period of 60 h. The question remains whether the fingerprint verification can be done in noisy conditions.

Das et al. (2014) tried to extract the features of not only the speaker but also the microphone. For the speaker, the authors recorded audio signals played by the speaker to extract the features. For the microphone, the authors observed audio clips recorded by the microphone. First, 15 well-known audio features are listed. Then the authors adopted a feature selection strategy to select some dominant features that can separate devices with the highest accuracy. The most dominant feature is mel-frequency cepstral coefficients (MFCCs), which concisely represent the power spectrum on a mel scale of frequency. According to experiments, the accuracy of identifying devices are always high enough that it is feasible to be applied in reality. The distance between speaker and recorder and sampling rate have an obvious impact on the accuracy, while training size and ambient background noise have little impact on the accuracy.

5.2. *Unique properties of internal hardware*

A Physically Unclonable Function (PUF) is the hardware analog of a cryptographic one-way function. Thus, a PUF must be easy to evaluate but hard to predict and also must be (practically) impossible to clone. However, unlike a normal function, PUFs are noisy. In recent years, PUF-based schemes have not only been suggested for the basic security tasks of tamper sensitive key storage or system identification, but also for more complex cryptographic protocols like the oblivious transfer (OT), bit commitment (BC), or key exchange (KE). In these works, so-called "Strong PUFs" are regarded as a new, fundamental cryptographic primitive of their own, comparable to the bounded storage model, quantum cryptography, or noise-based cryptography. Ruhrmair and van Dijk (2013) investigated the correct adversarial attack model and the actual security of such protocols. In particular, Ruhrmair and van Dijk (2013) define and compare different attack models. According to Ruhrmair and van Dijk (2013), the design of advanced cryptographic PUF protocols needs to be strongly reconsidered. Furthermore, it suggests that Strong PUFs require additional hardware properties in order to be broadly usable in such protocols: Firstly, they should ideally be "erasable", meaning that single PUF-responses can be erased without affecting other responses. If the area-efficient implementation of this feature turns out to be difficult, new forms of Controlled PUFs, such as Logically Erasable and Logically Reconfigurable PUFs, may suffice in certain applications. Secondly, PUFs should be "certifiable", meaning that one can verify that the PUF has been produced faithfully and has not been manipulated in any way afterward. The combined implementation of these features represents a pressing and challenging problem.

In Holcomb et al. (2009), the authors proposed a system, called FERNS, for fingerprint extraction and random numbers based on SRAM initial values. These values in SRAM at devices during start-up could be used as physical fingerprints to identify circuits and generate truly random numbers. The

Table 4 – Summary of the main approaches on physical context security.

Approach	Security Enhancement Area	Benefits	Issues
Absolute Location	Proximity	Accurate distance Existing devices (e.g. mobile GPS)	Needs location infrastructure (beacon, GPS, MNO) Both devices must be honest
Relative Location	Proximity	Resistant to more attack strategies	Needs special channel
Common Activity	Pairing/Proximity	Existing devices	Requires user assistance Only mobile devices
Environment Fingerprint	Pairing/Proximity	Existing devices	Active attacks not considered
Friendly Jamming	Pairing	Well suited to RFID Audio on existing devices	Radio hardware needs modification Jamming noise needs careful consideration
Auxiliary Channels	Pairing/Proximity	Use existing channels (sound, visual)	Channels really location limited? Complicates rather than prevent proximity attacks
Device Fingerprinting	Proximity	Inherent to devices	Low entropy Needs stable feature extraction Verifier needs additional hardware

strength of the FERNS method is that no dedicated circuits are required. Aside from the SRAM used to generate the fingerprints, the randomness extraction requires only a hash function or simple processing core capable of bit shift and bitwise XOR operations. FERNS has potential use across the spectrum of integrated circuit applications, ranging from low cost passively powered RFID tags and smart cards up through embedded caches on high-end devices

Approaches for remote physical device fingerprinting, or fingerprinting a physical device, or class of devices, remotely, and without the fingerprinted device's known cooperation are given in Kohno et al. (2005). This can be accomplished by exploiting small deviations in the hardware such as clock skews. This can be used to determine information about communicating devices on a network, possibly shifted in time or IP addresses, and this can determine if they are actually the same devices. Some example applications include computer forensics, tracking a physical device as it connects from different public access points, counting the number of communicating devices even when random IP addresses are used or determining if talking to real devices or virtual hosts.

Jana and Kasera (2010) proposed a method utilizing clock skew for identifying wireless access points (AP). Clock skew is the phenomenon that time in the devices is different than the actual time across different devices due to drift in the internal oscillators used to derive the clock. This method needs the verifier node, which produces the fingerprint, to receive a number of beacon frames belonging to the same AP. The authors also assumed that when these beacon frames are transmitting, the transmission rates are all the same. Then the authors tried to calculate the clock skew between the AP and verifier. T_i means the time on the time stamp in the ith frame and t_i means the receive time of ith frame on the node. O_i means the clock skew of the ith frame. The equation for calculating the clock skew is as follows:.

$$Oi = (Ti - T1) - (ti - t1) \tag{4}$$

The authors used frames sequence as x-axis and clock skew as y-axis. Then they used a least-square fitting method to estimate a line representing the clock skew of frames. This line is the identity of an AP. This method is a good idea but it should consider more about eliminating the impact of outliers when estimating the lines. Kohno et al. (2005) utilized a TCP time stamp to do similar work.

6. Conclusion

Pervasive systems require strong security mechanisms that allow for ad-hoc communication between devices of all capabilities. Although there are a wide array of cryptographic security solutions, these cannot provide adequate solutions for some scenarios encountered in this operating environment. In a truly ad-hoc environment devices have limited prior trust in each other. Devices need to establish a secure relationship, i.e. exchange key enabling cryptographic services, and also confirm that the device they are logically linking up with is the device physically present. We provided an overview of the main approaches offering these security services using physical context. A summary of the approaches and the services provided are given in Table 4.

We expect these physical context-based security supporting methods to continue in the following directions:

1. *Environment* Location is a good way to validate proximity. Absolute location is not suited to cities as high crowed buildings will weaken or block GPS signals. Relative location solves this problem. But as distance-bounding needs extremely precious time estimation, we need to find a more practical or stable way to measure the time. Both common activity and environment fingerprinting are user-friendly, but they need to think more about the error caused by the difference between generated keys.
2. *Communication* Friendly jamming is a simple way to transmit data securely, but the active attack is the next thing for researchers to face in this area. Auxiliary channels are user-friendly methods but errors caused by incorrect user operation is further research that needs to be concentrated on.

3. *Device* Both internal hardware and device fingerprinting have shown good results in the identification of devices. But a single feature is easy to be affected by outside factors, combining multiple features is a better way to get a robust result. Also building a database to store feature data is time consuming. To improve feature comparison methods should be focused on.

Acknowledgments

This work was funded by an Industrial Technology Fund grant (ITS/047/16) by the Industrial Technology Commision. Any opinions, findings, conclusions or recommendations expressed in this material/event (or by members of the project team) do not reflect the views of the Government of the Hong Kong Special Administrative Region, the Innovation and Technology Commission or the Panel of Assessors for the Innovation and Technology Support Programme of the Innovation and Technology Fund.

REFERENCES

Achard F, Savry O. A cross layer approach to preserve privacy in RFID ISO/IEC 15693 systems. Proceedings of RFID-TA. IEEE; 2012. p. 85–90.

Ali ST, Sivaraman V, Ostry D. Eliminating reconciliation cost in secret key generation for body-worn health monitoring devices. IEEE Trans Mob Comput 2014;13(12):2763–76.

Avoine G, Bingöl MA, Karda\cs S, Lauradoux C, Martin B. A framework for analyzing RFID distance bounding protocols. J Comput Secur 2011;19(2):289–317.

Balfanz D, Smetters DK, Stewart P, Wong HC. Talking to strangers: authentication in ad-hoc wireless networks. Proceedings of NDSS, 2002.

Bertoncini C, Rudd K, Nousain B, Hinders M. Wavelet fingerprinting of radio-frequency identification (RFID) tags. IEEE Trans Ind Electron 2012;59(12):4843–50.

Bonne Rasmussen K, Capkun S. Implications of radio fingerprinting on the security of sensor networks. Proceedings of third international conference on security and privacy in communications networks and the workshops, 2007. SecureComm 2007. IEEE; 2007. p. 331–40.

Brik V, Banerjee S, Gruteser M, Oh S. Wireless device identification with radiometric signatures. Proceedings of the 14th ACM international conference on Mobile computing and networking. ACM; 2008. p. 116–27.

Castelluccia C, Avoine G. Noisy tags: a pretty good key exchange protocol for RFID tags. Smart card research and advanced applications. Springer; 2006. p. 289–99.

Castelluccia C, Mutaf P. Shake them up!: a movement-based pairing protocol for CPU-constrained devices. Proceedings of the third international conference on mobile systems, applications, and services. ACM; 2005. p. 51–64.

Chagnaadorj O, Tanaka J. MimicGesture: secure device pairing with accelerometer-based gesture input. Ubiquitous information technologies and applications. Springer; 2013. p. 59–67.

Cheng B, Cui L, Jia W, Zhao W, Gerhard PH. Multiple region of interest coverage in camera sensor networks for tele-intensive care units. IEEE Trans Ind Inform 2016a;12(6):2331–41.

Cheng N, Lu N, Zhang N, Yang T, Shen XS, Mark JW. Vehicle-assisted device-to-device data delivery for smart grid. IEEE Trans Veh Technol 2016b;65(4):2325–40.

Danev B, Heydt-Benjamin TS, Capkun S. Physical-layer identification of RFID devices. Proceedings of the Usenix security symposium; 2009. p. 199–214.

Danev B, Zanetti D, Capkun S. On physical-layer identification of wireless devices. ACM Comput Surv 2012;45(1). 6:1–6:29

Das A, Borisov N, Caesar M. Do you hear what I hear?: fingerprinting smart devices through embedded acoustic components. Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. ACM; 2014. p. 441–52.

Dolui K, Datta SK. Comparison of edge computing implementations: fog computing, cloudlet and mobile edge computing. Proceedings of the Global Internet of Things Summit (GIoTS), 2017. IEEE; 2017. p. 1–6.

Echeverria S, Klinedinst D, Williams K, Lewis GA. Establishing trusted identities in disconnected edge environments. Proceedings of the , IEEE/ACM Symposium on Edge Computing (SEC). IEEE; 2016. p. 51–63.

Fang S, Liu Y, Shen W, Zhu H, Wang T. Virtual multipath attack and defense for location distinction in wireless networks. IEEE Trans Mob Comput 2017;16(2):566–80.

Fei H, Chouchang Y, Guang G, Radha P. A framework to securing RFID transmissions by varying transmitted reader's power. Proceedings of the radio frequency identification system security: RFIDsec'13 Asia Workshop. IOS Press; 2013. p. 57–68.

Gildas A, Muhammed AB, Ioana B, Srdjan Č, Gerhard H, Süleyman K, Chong HK, Cédric L, Benjamin M, Jorge M, et al. Security of distance- bounding: A survey. ACM Comput Surv., 2018.

Goodrich MT, Sirivianos M, Solis J, Tsudik G, Uzun E. Loud and clear: human-verifiable authentication based on audio. Proceedings of the 26th IEEE international conference on distributed computing systems, 2006. ICDCS 2006. IEEE; 2006. p. 10.

Gopala PK, Lai L, El Gamal H. On the secrecy capacity of fading channels. IEEE Trans Inf Theory 2008;54(10):4687–98.

Guillaume R, Winzer F, Czylwik A, Zenger CT, Paar C. Bringing PHY-based key generation into the field: an evaluation for practical scenarios. Proceedings of the IEEE 82nd vehicular technology conference (VTC Fall), 2015. IEEE; 2015. p. 1–5.

Halevi T, Saxena N. On pairing constrained wireless devices based on secrecy of auxiliary channels: the case of acoustic eavesdropping. Proceedings of the 17th ACM conference on computer and communications security. ACM; 2010. p. 97–108.

Hall J, Barbeau M, Kranakis E. Detecting rogue devices in bluetooth networks using radio frequency fingerprinting. Proceedings of the IASTED international conference on communications and computer networks. Citeseer, 2006.

Halperin D, Heydt-Benjamin TS, Ransford B, Clark SS, Defend B, Morgan W, Fu K, Kohno T, Maisel WH. Pacemakers and implantable cardiac defibrillators: software radio attacks and zero-power defenses. Proceedings of the IEEE symposium on security and privacy, 2008. SP 2008. IEEE; 2008. p. 129–42.

Hancke GP. Design of a secure distance-bounding channel for RFID. J Netw Comput Appl 2011;34(3):877–87.

Hancke GP. Distance-bounding for RFID: Effectiveness of âterrorist fraudâ in the presence of bit errors. Proceedings of the 2012 IEEE international conference on RFID-technologies and applications (RFID-TA). IEEE; 2012. p. 91–6.

Hancke GP, Kuhn MG. Attacks on time-of-flight distance bounding channels. Proceedings of the first ACM conference on wireless network security. ACM; 2008. p. 194–202.

Hancke GP, Mayes KE, Markantonakis K. Confidence in smart token proximity: relay attacks revisited. Comput Secur 2009;28(7):615–27.

Hermans J, Peeters R, Onete C. Efficient, secure, private distance bounding without key updates. Proceedings of the sixth ACM conference on security and privacy in wireless and mobile networks. ACM; 2013. p. 207–18.

Holcomb DE, Burleson WP, Fu K. Power-up SRAM state as an identifying fingerprint and source of true random numbers. IEEE Trans Comput 2009;58(9):1198–210.
Hu Q, Dinca LM, Hancke G. Device synchronisation: a practical limitation on reader assisted jamming methods for RFID confidentiality. Information Security Theory and Practice. Springer; 2015. p. 219–34.
Hu YC, Perrig A, Johnson DB. Packet leashes: a defense against wormhole attacks in wireless networks. Proceedings of the IEEE Societies 22nd annual joint conference of the IEEE computer and communications. INFOCOM 2003; 2003. p. 1976–86.
Jana S, Kasera SK. On fast and accurate detection of unauthorized wireless access points using clock skews. IEEE Trans Mob Comput 2010;9(3):449–62.
Jana S, Premnath SN, Clark M, Kasera SK, Patwari N, Krishnamurthy SV. On the effectiveness of secret key extraction from wireless signal strength in real environments. Proceedings of the 15th annual international conference on mobile computing and networking. ACM; 2009. p. 321–32.
Jin R, Du X, Deng Z, Zeng K, Xu J. Practical secret key agreement for full-duplex near field communications. Proceedings of the ninth ACM symposium on information, computer and communications security. ACM; 2014. p. 217–28.
Klein RW, Temple MA, Mendenhall MJ. Application of wavelet-based RF fingerprinting to enhance wireless network security. Commun Netw J 2009;11(6):544–55.
Kohno T, Broido A, Claffy KC. Remote physical device fingerprinting. IEEE Trans Depend Secure Comput 2005;2(2):93–108.
Kriara L, Alsup M, Corbellini G, Trotter M, Griffin JD, Mangold S. RFID shakables: pairing radio-frequency identification tags with the help of gesture recognition. Proceedings of the CoNEXT; 2013. p. 327–32.
Lai L, El Gamal H. The relay–eavesdropper channel: cooperation for secrecy. IEEE Trans Inf Theory 2008;54(9):4005–19.
Li Z, Yates R, Trappe W. Secret communication with a fading eavesdropper channel. Proceedings of the IEEE international symposium on information theory, ISIT 2007. IEEE; 2007. p. 1296–300.
Liu H, Wang Y, Yang J, Chen Y. Fast and practical secret key extraction by exploiting channel response. Proceedings of the INFOCOM, 2013. IEEE; 2013. p. 3048–56.
Liu H, Yang J, Wang Y, Chen Y. Collaborative secret key extraction leveraging received signal strength in mobile wireless networks. Proceedings of IEEE INFOCOM, 2012. IEEE; 2012. p. 927–35.
Ma D, Saxena N, Xiang T, Zhu Y. Location-aware and safer cards: Enhancing RFID security and privacy via location sensing. IEEE Trans Depend Secure Comput 2013;10(2):57–69.
Marino F, Paolini E, Chiani M. Secret key extraction from a UWB channel: analysis in a real environment. Proceedings of the 2014 IEEE international conference on Ultra-WideBand (ICUWB). IEEE; 2014. p. 80–5.
Mathur S, Miller R, Varshavsky A, Trappe W, Mandayam N. Proximate: proximity-based secure pairing using ambient wireless signals. Proceedings of the 9th international conference on mobile systems, applications, and services. ACM; 2011. p. 211–24.
Mathur S, Trappe W, Mandayam N, Ye C, Reznik A. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. Proceedings of the 14th ACM international conference on mobile computing and networking. ACM; 2008. p. 128–39.
Mayrhofer R. The candidate key protocol for generating secret shared keys from similar sensor data streams. Proceedings of the ESAS 2007. Springer-Verlag; 2007. p. 1–15.
Mayrhofer R, Gellersen H. Shake well before use: Intuitive and secure pairing of mobile devices. Mob Comput IEEE Trans 2009;8(6):792–806.
McCune JM, Perrig A, Reiter MK. Seeing-is-believing: using camera phones for human-verifiable authentication. Proceedings of the IEEE symposium on Security and privacy, 2005. IEEE; 2005. p. 110–24.
Mirzadeh S, Cruickshank H, Tafazolli R. Secure device pairing: a survey. IEEE Commun Surv Tutor 2014;16(1):17–40.
Nandakumar R, Chintalapudi KK, Padmanabhan V, Venkatesan R. Dhwani: secure peer-to-peer acoustic NFC. Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM. ACM; 2013. p. 63–74.
Periaswamy SCG, Thompson DR, Di J. Fingerprinting RFID tags. IEEE Trans Depend Secure Comput 2011;8(6):938–43.
Periaswamy SCG, Thompson DR, Romero HP, Di J. Fingerprinting radio frequency identification tags using timing characteristics. Proceedings of the workshop on RFID Security-RFIDsec Asia, 2010.
Perkovic T, Cagalj M, Mastelic T, Saxena N, Begusic D. Secure initialization of multiple constrained wireless devices for an unaided user. IEEE Trans Mob Comput 2012;11(2):337–51.
Pinto PC, Barros J, Win MZ. Wireless physical-layer security: the case of colluding eavesdroppers. Proceedings of the IEEE international symposium on information theory, 2009. ISIT 2009. IEEE; 2009. p. 2442–6.
Prasad R, Saxena N. Efficient device pairing using "Human-comparable" synchronized audiovisual patterns. Proceedings of the sixth international conference on applied cryptography and network security ACNS'08; 2008. p. 328–45.
Premnath SN, Jana S, Croft J, Gowda PL, Clark M, Kasera SK, Patwari N, Krishnamurthy SV. Secret key extraction from wireless signal strength in real environments. IEEE Trans Mob Comput 2013;12(5):917–30.
Quach Q, Nguyen N, Dinh T. Secure authentication for mobile devices based on acoustic background fingerprint. Proceedings of the knowledge and systems engineering. Springer; 2014. p. 375–87.
Ranganathan A, Tippenhauer NO, Skoric B, Singelée D, Capkun S. Design and Implementation of a terrorist fraud resilient distance bounding system. In: Foresti S, Yung M, Martinelli F, editors. ESORICS. Springer; 2012. p. 415–32.
Rasmussen KB, Capkun S. Realization of RF distance bounding. Proceedings of the USENIX security symposium; 2010. p. 389–402.
Rivest RL, Shamir A. How to expose an eavesdropper. Commun ACM 1984;27(4):393–4.
Ruhrmair U, van Dijk M. PUFs in security protocols: attack models and security evaluations. Proceedings of the IEEE computer society symposium on security and privacy; 2013. p. 286–300.
Savry O, Pebay-Peyroula F, Dehmas F, Robert G, Reverdy J. RFID Noisy Reader How to Prevent from Eavesdropping on the Communication?. Proceedings of the ninth international workshop on cryptographic hardware and embedded systems - CHES 2007, Vienna, Austria, September 10–13, 2007. Springer; 2007. p. 334–45.
Saxena N, Ekberg JE, Kostiainen K, Asokan N. Secure device pairing based on a visual channel. Proceedings of the IEEE symposium on security and privacy, 2006. IEEE, 2006.
Saxena N, Ekberg JE, Kostiainen K, Asokan N. Secure device pairing based on a visual channel: design and usability study. IEEE Trans Inf Forensics Secur 2011a;6:28–38.
Saxena N, Uddin MB, Voris J. Universal device pairing using an auxiliary device. Proceedings of the fourth symposium on uable privacy and security. ACM; 2008. p. 56–67.
Saxena N, Uddin MB, Voris J, Asokan N. Vibrate-to-unlock: Mobile phone assisted user authentication to multiple personal RFID

tags. Proceedings of the IEEE international conference on pervasive computing and communications (PerCom), 2011. IEEE; 2011b. p. 181–8.
Schurmann D, Sigg S. Secure communication based on ambient audio. IEEE Trans Mob Comput 2013;12(2):358–70.
Shen W, Ning P, He X, Dai H. Ally friendly jamming: how to jam your enemy and maintain your own wireless connectivity at the same time. Proceedings of the IEEE symposium on security and privacy (SP), 2013. IEEE; 2013. p. 174–88.
Soriente C, Tsudik G, Uzun E. BEDA: button-enabled device pairing, International Workshop on Security for Spontaneous Interaction. UbiComp Workshop Proceedings 2007.
Soriente C, Tsudik G, Uzun E. HAPADEP: human-assisted pure audio device pairing. Proceedings of the information security. Springer; 2008. p. 385–400.
Stajano F, Anderson R. The resurrecting duckling: security issues for Ad-Hoc wireless networks. Proceedings of the security protocols. Springer; 2000. p. 172–82.
Stajano F, Wong FL, Christianson B. Multichannel protocols to prevent relay attacks. Proceedings of the financial cryptography and data security. Springer; 2010. p. 4–19.
Studer A, Perrig A. Mobile user location-specific encryption (MULE): using your office as your password. Proceedings of the third ACM conference on wireless network security WiSec '10; 2010. p. 151–62.
Tang X, Liu R, Spasojevic P, Poor HV. The Gaussian wiretap channel with a helping interferer. Proceedings of the IEEE international symposium on information theory, 2008. ISIT 2008. IEEE; 2008. p. 389–93.
Tang X, Liu R, Spasojevic P, Poor HV. Interference assisted secret communication. IEEE Trans Inf Theory 2011;57(5):3153–67.
Technology V. How much Data will the Internet of Things (IoT) generate by 2020? 2017. https://www.versatek.com/blog/how-much-data-will-the-internet-of-things-iot-generate-by-2020/
Tekin E, Yener A. The general Gaussian multiple-access and two-way wiretap channels: achievable rates and cooperative jamming. IEEE Trans Inf Theory 2008;54(6):2735–51.
Tippenhauer NO, Malisa L, Ranganathan A, Capkun S. On limitations of friendly jamming for confidentiality. Proceedings of the IEEE symposium on security and privacy. IEEE Computer Society; 2013. p. 160–73.
Ureten O, Serinken N. Wireless security through RF fingerprinting. Electr Comput Eng Can J 2007;32(1):27–33.
Vaudenay S. Secure communications over insecure channels based on short authenticated strings. Proceedings of the advances in cryptology–CRYPTO 2005. Springer; 2005. p. 309–26.
Vilela JP, Bloch M, Barros J, McLaughlin SW. Wireless secrecy regions with friendly jamming. Secur IEEE Trans Inf Forensics 2011;6(2):256–66.
Wei Y, Zeng K, Mohapatra P. Adaptive wireless channel probing for shared key generation based on PID controller. IEEE Trans Mob Comput 2013;12(9):1842–52.
Wilhelm M, Martinovic I, Schmitt JB. Secure key generation in sensor networks based on frequency-selective channels. IEEE J Sel Areas Commun 2013;31(9):1779–90.
Wilson R, Tse D, Scholtz RA. Channel identification: Secret sharing using reciprocity in ultrawideband channels. IEEE Transactions on Information Forensics and Security 2007;2(3):364–75.
Wyner AD. The wire-tap channel. Techn J Bell Syst 1975;54(8):1355–87.
Xi W, Li XY, Qian C, Han J, Tang S, Zhao J, Zhao K. KEEP: Fast secret key extraction protocol for D2D communication. Proceedings of the IEEE 22nd international symposium of Quality of Service (IWQoS), 2014. IEEE; 2014. p. 350–9.
Yang A, Pagnin E, Mitrokotsa A, Hancke G, Wong DS. Two-hop distance-bounding protocols: keep your friends close. IEEE Trans Mob Comput 2018;17(7):1723–36.
Zanetti D, Danev B, Others. Physical-layer identification of UHF RFID tags. Proceedings of the sixteenth annual international conference on mobile computing and networking. ACM; 2010. p. 353–64.
Zeng K, Wu D, Chan A, Mohapatra P. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. Proceedings of the INFOCOM, 2010. IEEE; 2010. p. 1–9.
Zenger CT, Chur MJ, Posielek JF, Paar C, Wunder G. A novel key generating architecture for wireless low-resource devices. Proceedings of the 2014 international workshop on Secure Internet of Things (SIoT). IEEE; 2014. p. 26–34.
Zhang B, Zhan Q, Wang J, Ren K, Wang C, Ma D. PriWhisper: enabling keyless secure acoustic communication for smartphones. IACR Cryptol ePrint Arch 2013;2013:581.
Zhang J, Duong TQ, Marshall A, Woods R. Key generation from wireless channels: a review. IEEE Access 2016;4:614–26.
Zhang J, Woods R, Marshall A, Duong TQ. Verification of key generation from individual OFDM subcarrier's channel response. Proceedings of the IEEE Globecom workshops (GC Wkshps), 2015. IEEE; 2015. p. 1–6.
Zhou X, Tao M, Kennedy RA. Cooperative jamming for secrecy in decentralized wireless networks. Proceedings of the 2012 IEEE international conference on communications (ICC). IEEE; 2012. p. 2339–44.
Zhou Z, Diao W, Liu X, Zhang K. Acoustic fingerprinting revisited: generate stable device id stealthily with inaudible sound. Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. ACM; 2014. p. 429–40.
Zhu X, Xu F, Novak E, Tan CC, Li Q, Chen G. Extracting secret key from wireless link dynamics in vehicular environments. Proceedings of the IEEE INFOCOM, 2013. IEEE; 2013. p. 2283–91.

Qiao HU received the BS degree from Hunan University in 2011 and the MSc degree from Wuhan University in 2013. He got his Ph.D. degree in the Department of Computer Science, City University of Hong Kong in 2017. Now he works as a research associate in City University of Hong Kong. His research interests include RFID security and privacy, cloud computing and wireless communication security.

Jingyi ZHANG received the B.Eng. degree in computer science from Zhengzhou University in 2015 and the M.Eng. degree in Telecommunications from Hong Kong University of Science and Technology in 2017. Then she worked in the Spanish National Research Council (CSIC) as a research assistant for half a year and now she works in City University of Hong Kong as a research assistant. Her research interests include physical layer security, RFID, and wireless communication.

Aikaterini Mitrokotsa is an Associate Professor in the Department of Computer Science and Engineering, Chalmers University of Technology. Previously, she held positions as a visitor professor with ETHZ and the Tokyo Institute of Technology. Her main research interests include the information security, privacy-preservation, authentication protocols, and provable security. She has been awarded the Young Researcher Grant from the Swedish Research Council, the Rubicon Research Grant by NWO, and a Marie Curie Intra European Fellowship.

Gerhard P. HANCKE is an Assistant Professor with City University of Hong Kong since 2013. He received B.Eng and M.Eng. degrees in Computer Engineering from the University of Pretoria (South Africa), in 2002 and 2003 respectively, and a Ph.D. degree in Computer Science from the University of Cambridge (UK) in 2008. His research interests include applications and system security of industrial IoT and cyber-physical systems. His research interests are system security, embedded platforms and distributed sensing applications especially with regards to industrial IoT and cyber-physical systems.