A Cross-Layer Key Establishment Model for Wireless Devices in Cyber-Physical Systems

Yuexin Zhang
Centre for Cyber Security Research
Deakin University
Geelong, VIC 3220, Australia
yuexinz@deakin.edu.au

Yang Xiang
Centre for Cyber Security Research
Deakin University
Geelong, VIC 3220, Australia
yang.xiang@deakin.edu.au

Xinyi Huang*
School of Mathematics and Computer Science
Fujian Normal University
Fuzhou, 350108, China
xyhuang@fjnu.edu.cn

ABSTRACT

Wireless communications in Cyber-Physical Systems (CPS) are vulnerable to many adversarial attacks such as eavesdropping. To secure the communications, secret session keys need to be established between wireless devices. In existing symmetric key establishment protocols, it is assumed that devices are pre-loaded with secrets. In the CPS, however, wireless devices are produced by different companies. It is not practical to assume that the devices are pre-loaded with certain secrets when they leave companies. As a consequence, existing symmetric key establishment protocols cannot be directly implemented in the CPS. Motivated by these observations, this paper presents a cross-layer key establishment model for heterogeneous wireless devices in the CPS. Specifically, by implementing our model, wireless devices extract master keys (shared with the system authority) at the physical layer using ambient wireless signals. Then, the system authority distributes secrets for devices (according to an existing symmetric key establishment protocol) by making use of the extracted master keys. Completing these operations, wireless devices can establish secret session keys at higher layers by calling the employed key establishment protocol. Additionally, we prove the security of the proposed model. We analyse the performance of the new model by implementing it and converting existing symmetric key establishment protocols into cross-layer key establishment protocols.

Keywords

Key establishment; security; cross-layer; wireless devices; cyber-physical systems

1. INTRODUCTION

Nowadays, an increasing number of devices are equipped with wireless interfaces. Additionally, it is estimated that 50 to 100 billion devices will be wirelessly connected to the Internet of Things/Internet of Everything (IoT/IoE) by 2020 [21]. In practical applications, a multitude of wireless devices have already been deployed and they form the Cyber-Physical Systems (CPS). Specifically, the CPS devices are heterogeneous, and they constitute interconnected systems [16] (Figure 1 shows applications of the CPS). According to [6], however, the CPS becomes vulnerable to many malicious attacks. For instance, in 2006, an attacker compromised a computer at a water filtering plant in Pennsylvania, and the compromised computer was used as the attacker's distribution system for spam and pirated software [6]. Besides, Several industrial infrastructures in Queensland and Australia were attacked by the Stuxnet. After suffering at least three years' attacks, the Stuxnet was discovered in 2010 [27]. Recently, Dyn experienced two distributed denial of service attacks on its DNS servers, and a number of websites, such as Twitter, Github, Vox, Spotify, and Netflix, went down on October 21, 2016.

To fight against potential attacks in wireless communications, many security protocols have been proposed, including location protocols [3, 28], intrusion detection protocols [15, 22], secure routing protocols [13, 14], authentication proto-

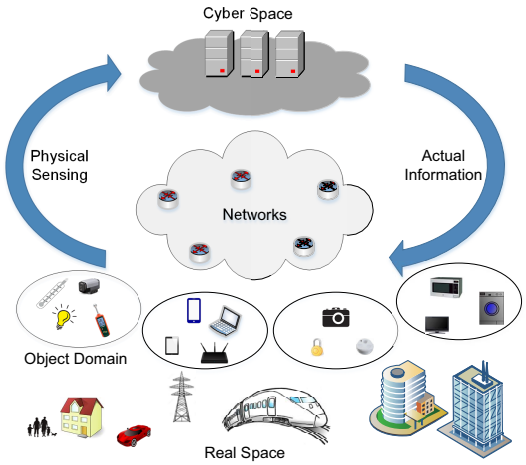

Figure 1: Applications of the CPS with interconnecting boundary between cyber and object domain [2].

*Corresponding Author

CPSS'17, April 02 2017, Abu Dhabi, United Arab Emirates

DOI: http://dx.doi.org/10.1145/3055186.3055187

cols [18, 8], and key establishment protocols [10, 7, 5, 17, 4, 9]. As a fundamental security countermeasure, key establishment has been extensively and intensively studied, and many key establishment protocols have been proposed at higher layers. Specifically, these key establishment protocols can be classified into two main types, i.e., asymmetric key establishment protocols and symmetric key establishment protocols. In asymmetric key establishment protocols, devices need to execute costly computation operations, such as the modular exponentiation operations. Recall that wireless devices in the CPS may be energy-constraint devices (e.g., sensor nodes). Thus, in the CPS, the energy intensive asymmetric key establishment protocols are excluded. In symmetric key establishment protocols (such as the key pre-distribution protocols), it is assumed that devices are pre-loaded with secrets. Making use of the pre-loaded secrets, two devices can establish a secret session key with certain probability.

However, existing symmetric key establishment protocols cannot be directly implemented in the CPS scenario when the secret-sharing assumption cannot be met. For instance, wireless devices in the CPS are heterogeneous ones, and they are produced by different companies. Thus, it is not practical to assume that the devices are pre-loaded with certain secrets when they leave companies. Motivated by these observations, we aim to design a key establishment model for wireless devices such that existing symmetric key establishment protocols can be directly implemented in the CPS.

Looking at our modern lives, we are drowned in kinds of wireless signals, such as $3G$ and $4G$ signals, TV signals, and Wi-Fi signals. Recently, there is an increasing interest in extracting secret keys by taking advantage of the transmitted signals. In the typical multipath environments, the wireless channel between two devices, e.g., Alice and Bob, experiences a time-varying, stochastic fading between the transmitted and received signals. Specifically, the fading is unique, location-specific and reciprocal. Namely, it is invariant within the channel coherence time whether the signals are transmitted from Alice to Bob or from Bob to Alice. In wireless communications, the channel coherence time is a statistical measure of the time duration over which the channel impulse response is essentially invariant. Additionally, it is widely recognised that wireless devices can extract secret keys from ambient wireless signals (please refer to Subsections 2.2 and 3.2 for details).

In these key extraction protocols (proposed at the physical layer using wireless fading channels), however, some issues still remain unsettled. For example, the key generation rate needs to be improved, and a dynamic environment is needed to provide sufficient entropy (please refer to Subsections 2.2 and 3.2 for details). Thus, it is impractical to extract session keys using the wireless fading channel when a large number of session keys need to be established.

Our Contribution. In the CPS, wireless devices need to establish session keys for the purpose of securing the communications. In practice, however, wireless devices in the CPS are produced by different companies. Thus, it is not practical to assume that the devices are pre-loaded with certain secrets when they leave companies. As a result, existing symmetric key establishment protocols cannot be directly implemented in the CPS. Moreover, it is not practical to extract session keys using the wireless fading channel when a large number of session keys need to be established. However, it should be a reasonable idea to alleviate these problems by cooperatively utilising the characteristics of these two types of key establishment protocols. Motivated by these observations, in this paper, we design a cross-layer key establishment model for wireless devices such that existing symmetric key establishment protocols can be directly implemented in the CPS. Specifically, the proposed model possesses the following properties:

1. Our key establishment model is designed for assisting wireless devices, who do not pre-share any secrets, to establish secret session keys. Specifically, the proposed model is a cross-layer design. Namely, wireless devices extract master keys (shared with the system authority) at the physical layer when joining the CPS. Making use of the extracted master keys, the system authority distributes secrets for devices (according to an existing symmetric key establishment protocol). Completing these operations, wireless devices can establish secret session keys at higher layers by calling the employed key establishment protocol; and

2. We prove the security of the cross-layer key establishment model. Additionally, we analyse the performance of the model by implementing it and converting existing symmetric key establishment protocols into cross-layer key establishment protocols. The analysis illustrates that employing the proposed model, existing symmetric key establishment protocols can be directly implemented by wireless devices in the CPS.

Organization of The Paper. The remainder of this paper is organized as follows. In the next section, we review the related work. Section 3 introduces the preliminaries required in this paper. Then, the proposed model is presented in Section 4, and its security and performance analysis are provided in Section 5 and Section 6, respectively. In Section 7, we conclude this paper.

2. RELATED WORK

This section reviews the related key establishment protocols, i.e., the symmetric key establishment protocols (proposed at higher layers) and the key extraction protocols using the wireless fading channel (proposed at the physical layer).

2.1 Symmetric Key Establishment Protocols

Until now, many symmetric key establishment protocols have been proposed at higher layers. In this subsection, we review several types of these protocols. Specifically, in Section 6, we will convert these reviewed protocols into cross-layer key establishment protocols by implementing our proposed model.

A random key pre-distribution protocol was presented by Eschenauer and Gligor in [10], and Chan et al. improved it in [7] by designing a q-composite random key pre-distribution (q-KP) protocol. Specifically, there are three phases in [7], i.e., the key pre-distribution phase, the shared key discovery phase, and the session key establishment phase. In the key pre-distribution phase, the system authority generates a set of secrets keys. For each sensor node, the system authority randomly chooses m keys. It is assumed that the chosen keys are loaded into the nodes via secure channels or

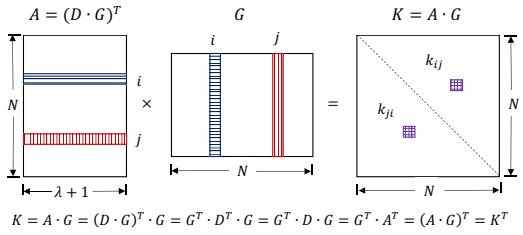

Figure 2: The core idea of Du et al.'s matrix-based key establishment protocol [9].

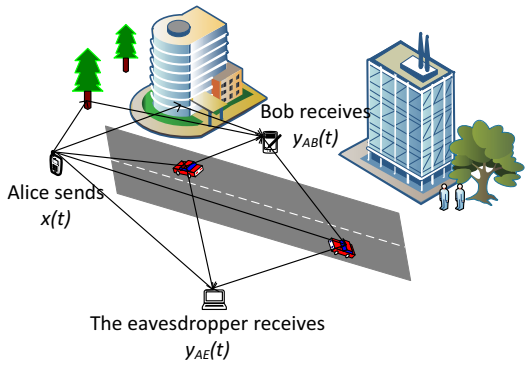

Figure 3: An example of extracting secret bits using the wireless fading channel.

when the system authority is off-line. In the shared key discovery phase, each node broadcasts the identifiers of stored keys and find the common keys it shares with its neighbors. Then, the node can establish session keys with its neighbor nodes (when they share at least q keys).

In [5], a polynomial based key pre-distribution protocol (also known as Blundo's protocol) was proposed. Specifically, in [5], a randomly generated t-degree polynomial $f(x,y) = \sum_{i,j=0}^{t} a_{ij}x^i y^j$ is employed. The generated polynomial satisfies the property $f(x,y) = f(y,x)$. Besides, Liu and Ning improved the protocol of [5] and proposed a polynomial pool based key establishment (PKE) protocol [17]. There are three phase in [17], i.e., the setup phase, the key pre-distribution phase, and the key establishment phase. In the setup phase, the system authority generates a set $\mathcal{F}$ of bivariate t-degree polynomials over the finite field $GF(q)$. The identifier ID_i is used to identify the i^{th} polynomial $f_i(x,y) = \sum_{i,j=0}^{t} a_{ij}x^i y^j$, where $f_i(x,y) \in \mathcal{F}$. In the key pre-distribution phase, the system authority randomly chooses a subset $\mathcal{F}_i$ of polynomials (from the polynomial pool $\mathcal{F}$, i.e., $\mathcal{F}_i \subseteq \mathcal{F}$) for each sensor node. It is assumed that the shares of chosen polynomials are distributed to each node via secure channels or when the system authority is off-line. In the key establishment phase, two nodes i and j can compute a session key by exchanging the stored polynomials' identifiers IDs and discovering the shared polynomial(s).

In [4], a matrix-based key establishment protocol was presented by Blom. The protocol ensures that any two nodes can establish a secret session key by exchanging some public information. Then, Du et al. employed the multiple key-spaces idea and improved the protocol [4] by designing a new matrix-based key establishment (MKE) protocol in [9]. Specifically, there are two phases in [9], i.e., the key pre-distribution phase and the key agreement phase. In the key pre-distribution phase, the system authority generates a $(\lambda+1) \times N$ public matrix G and ω secret symmetric matrices $D_1, D_2, \ldots, D_\omega$, and computes matrices $A_i = (D_i \cdot G)^T$ for each D_i. Then, the system authority randomly selects τ A_is and loads the k^{th} node with the k^{th} row of each selected A_i and the k^{th} key seed of G. It is assumed that the selected data is loaded into the nodes via secure channels or when the system authority is off-line. In the key agreement phase, two nodes can establish a session key with certain probability by broadcasting the identifiers of stored matrices (i.e., two nodes can establish a session key when they are loaded with rows from the same matrices A_is, as shown in Figure 2).

In these symmetric key establishment protocols, it is assumed that secrets are pre-loaded into the devices via secure channels or when the system authority is off-line. Thus, these protocols cannot be directly implemented in the CPS when the assumption cannot be met. Motivated by this observation, we aim to design a key establishment model such that these protocols can be directly implemented in the CPS when the assumption fails.

2.2 Key Extraction Protocols Using the Wireless Fading Channel

In the past two decades, many key extraction protocols were proposed by taking advantage of the wireless fading channel's characteristics. Specifically, in the typical multipath environments, the wireless channel between two users, e.g., Alice and Bob, experiences a time-varying, stochastic mapping between the transmitted and received signals. This mapping (commonly termed fading) is unique, location-specific and reciprocal. Namely, the fading is invariant within the channel coherence time whether the signals are transmitted from Alice to Bob or vice-versa. In wireless communications, the coherence time is a statistical measurement of the time duration over which the channel impulse response is essentially invariant. According to the communication theory, the fading decorrelates over distances of the order of half a wavelength ($\lambda/2$). Namely, the signals transmitted between Alice and Bob and the signals transmitted between Alice (or Bob) and the eavesdropper experience independent fading, when the eavesdropper is at least $\lambda/2$ away from Alice and Bob. In other words, the eavesdropper cannot obtain any useful information as long as it is $\lambda/2$ away from Alice and Bob. Taking the IEEE standard 802.15.4 as an example. The 802.15.4 specifies the frequency bands of the physical layer [1], i.e., 868 MHz, 915 MHz, and 2400 MHz. Thus, we can evaluate that $\lambda/2 \approx 17.28$ cm when the frequency band is 868 MHz; $\lambda/2 \approx 16.39$ cm when the frequency band is 915 MHz; and $\lambda/2 \approx 6.25$ cm when the frequency band is 2400 MHz.

To facilitate understanding, Figure 3 shows an example. In this example, we assume that Alice and Bob want to extract a secret key using the wireless fading channel. Firstly, Alice sends a sinusoidal signal $x(t) = A\sin(w_c t + \varphi_0)$ to Bob. Here A is the amplitude, w_c is the angular frequency, and φ_0 is the initial phase. Due to the multipath environment, noise, and/or mobile environment, the signals received at Bob and the eavesdropper are modulated by independent fading channels (as shown in Figure 3, we assume that the eavesdropper is more than $\lambda/2$ away from Alice and Bob).

We denote by $y_{AB}(t)$ and $y_{AE}(t)$ the signals received at Bob and the eavesdropper, and they can be written as:

$$\begin{aligned} y_{AB}(t) &= (A + A_{AB})\sin(w_c t + \varphi_0 + \varphi_{AB}) + n_{AB}(t), \\ y_{AE}(t) &= (A + A_{AE})\sin(w_c t + \varphi_0 + \varphi_{AE}) + n_{AE}(t). \end{aligned}$$

Here, A_{AB} and A_{AE} are the modulated amplitudes, and they are functions of path loss and shadowing; φ_{AB} and φ_{AE} are the deviated phases, and they depend on delay, Doppler, and carrier offset. $n_{AB}(t)$ and $n_{AE}(t)$ denote the additive white Gaussian noise. Receiving signal $y_{AB}(t)$, Bob replies Alice with the signal $x(t) = A\sin(w_c t + \varphi_0)$ in the coherence time. Similarly, the signal received by Alice and the eavesdropper are $y_{BA}(t)$ and $y_{BE}(t)$, and they can be written as:

$$\begin{aligned} y_{BA}(t) &= (A + A_{BA})\sin(w_c t + \varphi_0 + \varphi_{BA}) + n_{BA}(t), \\ y_{BE}(t) &= (A + A_{BE})\sin(w_c t + \varphi_0 + \varphi_{BE}) + n_{BE}(t). \end{aligned}$$

If the above signals are transmitted in the coherence time, then we have the modulated amplitudes $A_{AB} = A_{BA}$ and the deviated phases $\varphi_{AB} = \varphi_{BA}$. If the eavesdropper is at least $\lambda/2$ away from Alice and Bob, it cannot extract any useful secrets by taking advantage of the received signals $y_{AE}(t)$ and $y_{BE}(t)$. Namely, A_{AE} and A_{AB}, A_{BE} and A_{BA}, φ_{AE} and φ_{AB}, φ_{BE} and φ_{BA} are statistically independent as long as the eavesdropper is more than $\lambda/2$ away from Alice and Bob. In practice, some other technologies, such as quantization, information reconciliation, and privacy amplification, need to be employed in order to ensure that Alice and Bob can correctly extract a secret key [30] (from the extracted randomness A_{AB} and A_{BA}, φ_{AB} and φ_{BA}).

Until now, many key extraction protocols have been proposed at the physical layer by taking advantage of characteristics of the wireless fading channel, such as the Received Signal Strength (RSS) and Channel Impulse Response (CIR). In [20, 12, 29, 23], for instance, the attenuation of amplitude was employed to extract secret keys. More specifically, in Mathur et al.'s protocol [20], two devices can evaluate the envelope of multipath fading channel between them by probing a fixed test frequency. Then, they can obtain secret bits by quantifying the evaluation. Additionally, to validate their algorithm, the 802.11a packet preamble was used on a FPGA-based 802.11 platform. The experiment shows that their algorithm can achieve key extraction rates of 1 bit/sec in the indoor wireless environment. By exploiting technologies, e.g., quantization, information reconciliation, and privacy amplification, Jana et al. in [12] evaluated the efficiency of secret key extraction using RSS variations in different environments and settings. Besides, Vehicle-to-Infrastructure and Vehicle-to-Vehicle communication keys were extracted in [29] by using the attenuation of envelope. In [23], an environment adaptive secret key extraction protocol was proposed.

The deviation of phase (or phase offset) also be used to extract secret bits. For example, it is used to extract secret keys in [31, 25, 26]. Specifically, in order to accelerate the key bit generation rate, multiple-antenna diversities were exploited in [31]. In [31], Zeng et al. implemented their key extraction algorithm on off-the-shelf 802.11n multiple-antenna devices. The analysis shows that using laptops with three antennas, protocol in [31] can increase the key generation rate by more than 4 times over single-antenna systems. In [25], the uniformly distributed phase information of channel responses (under narrowband multipath fading models) was utilized to extract pairwise keys and group keys. Besides, a cooperative key generation protocol was proposed in [26] with the aid of relay node(s). In [19], Mathur et al. designed a novel key extraction protocol using the ambient wireless signals. The basic principle employed in [19] is similar to that of principle used in [20, 12, 29, 23, 31, 25, 26], and we will introduce the protocol [19] in Subsection 3.2.

In practice, however, some issues exist in these key extraction protocols, and it still remains unsatisfactory. For example, the key generation rate needs to be improved, and a dynamic environment is needed in order to provide sufficient entropy. Thus, it is not practical for wireless devices to extract session keys using the wireless fading channel (when a large number of session keys need to be established). Implementing our model, each device only extracts a master key (shared with the system authority) when it joins the CPS. Then, the system authority distributes secrets for devices (according to an existing symmetric key establishment protocol) by making use of the extracted master keys. Completing these operations, two wireless devices can establish a secret session key at higher layers by calling the employed key establishment protocol.

3. PRELIMINARIES

Before presenting our cross-layer key establishment model, in this section, we introduce the preliminaries required in this paper.

3.1 Security Model

This subsection reviews the security model of our cross-layer key establishment design. Specifically, we assume that N devices in the CPS are wirelessly communicate with each other. We denote by $\mathcal{D}$ the set of N devices. For the i^{th} device D_i $(i = 1, 2, \ldots, N)$, we have $D_i \in \mathcal{D}$. Additionally, we assume that the system authority is a trusted entity. In our model, the system authority is used to generate secrets according to the input security parameter 1^k and the employed key establishment protocol.

Adversarial model. We consider the adversary who aims to compute and obtain the session key established between two noncompromised devices. Specifically, we assume that the communications can be eavesdropped by the adversary. Namely, the passive adversary eavesdrops the communications and conducts sophisticated data analysis. Moreover, we assume that in order to obtain the session key, the active adversary replays and tampers the transmitted messages, and inserts bogus messages. The cross-layer key establishment model is a secure model if the adversary has the probability at most

$$P_{AKE,\mathcal{A},P}^{compromise}(k) \leq \varepsilon(k)$$

to disclose the established session key between two benign devices, where $\varepsilon(k)$ is a negligible probability.

3.2 The Key Extraction Protocol Using Ambient Wireless Signals

Mathur et al. in [19] investigated that the ambient wireless signals (such as TV signals, radio signals, and WiFi signals) can be used to extract secret bits. The basic principles employed in [19] is similar to that of principles in [20, 12, 29, 23, 31, 25, 26]. To facilitate understanding, Figure 4 shows the core idea of Mathur et al.'s key extraction algorithm [19].

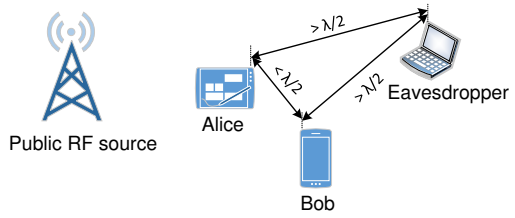

Figure 4: An example of extracting secret bits using ambient wireless signals.

To simplify the descriptions, in Figure 4 we assume that there is only one public RF source. For instance, it may be a radio station tower. Then, we assume that the public RF source (S) broadcasts the sinusoidal signal $x(t) = A\sin(w_c t + \varphi_0)$. Due to the multipath environment, noise, and/or mobile environment, the signal received at Alice, Bob, and the eavesdropper are $y_{SA}(t)$, $y_{SB}(t)$, and $y_{SE}(t)$, and they can be written as:

$$y_{SA}(t) = (A + A_{SA})\sin(w_c t + \varphi_0 + \varphi_{SA}) + n_{SA}(t),$$
$$y_{SB}(t) = (A + A_{SB})\sin(w_c t + \varphi_0 + \varphi_{SB}) + n_{SB}(t),$$
$$y_{SE}(t) = (A + A_{SE})\sin(w_c t + \varphi_0 + \varphi_{SE}) + n_{SE}(t).$$

As shown in Figure 4, the modulated amplitudes $A_{SA} = A_{SB}$ and the deviated phases $\varphi_{SA} = \varphi_{SB}$ if Alice and Bob are within $\lambda/2$ distance. However, the modulated amplitudes A_{SE} and A_{SA}, A_{SE} and A_{SB}, and the deviated phases φ_{SE} and φ_{SA}, φ_{SE} and φ_{SB} are statistically independent as long as the eavesdropper is more than $\lambda/2$ away from Alice and Bob. Namely, the eavesdropper cannot obtain any useful secrets (by making use of its received signals $y_{SE}(t)$) when it is more than $\lambda/2$ away from Alice and Bob.

Using the extracted measurements (i.e., the modulated amplitudes and deviated phases), Alice and Bob quantize them and end up with n-bit sequences. In order to extract a secret key, other technologies, including reconciliation, privacy amplification, and list-encoding, need to be employed by Alice and Bob. Please refer to [19] for details. Furthermore, Mathur et al. in [19] evaluate their algorithm using an experimental prototype built on top of GNUradio. Specifically, some real RF signals, e.g., the TV signals at 584.31 MHz ($\lambda/2 = 0.26$ m), the FM-radio broadcast band at 98 MHz ($\lambda/2 = 1.53$ m) in the NY/NJ area, US, are employed in their experiment. The experiment shows that a stationary Alice and Bob can extract a new bit from the TV signal and the FM signal every 0.27 seconds and 1.25 seconds, respectively. Taking the AES-128 as an example, it needs around 34.56 seconds (when $f = 584.31$ MHz) and 160.00 seconds (when $f = 98$ MHz) to extract a key with 128 bits. Thus, it becomes impractical to extract session keys using the ambient wireless signals when a large number of session keys needs to be established. In this paper, we design a cross-layer key establishment model such that wireless devices can establish session keys efficiently when a large number of session keys need to be established.

4. A CROSS-LAYER KEY ESTABLISHMENT MODEL FOR WIRELESS DEVICES IN THE CPS

This section presents the details of our cross-layer key establishment model. Specifically, the model is designed based on the following observations. In existing symmetric key establishment protocols, it is assumed that the system authority pre-distributes secrets for devices via secure channels or when it is off-line. In certain applications (such as in the CPS), the assumption cannot be met. As a result, the existing symmetric key establishment protocols cannot be directly implemented in these applications. Furthermore, it is impractical to extract session keys using the ambient wireless signals when a large number of session keys need to be established. However, it should be a reasonable idea to alleviate these problems by utilising the characteristics of these two types of key establishment protocols cooperatively. Thus, this section presents a key establishment model such that existing symmetric key establishment protocols can be directly implemented in the CPS. Specifically, the model is a cross-layer design. Namely, each device only extracts a master key (shared with the system authority) at the physical layer using the ambient wireless signals. Then, the system authority distributes secrets for devices (according to an existing symmetric key establishment protocol). Completing these operations, devices can establish session keys at higher layers by calling the employed key establishment protocol.

4.1 Overview

Our cross-layer key establishment model consists of four phases:

- **Initialization**. In this phase, the system authority generates system parameters, such as the secrets and a public hash function $H(x)$.
- **Master Key Extraction**. In this phase, devices extract master keys (shared with the system authority) at the physical layer.
- **Secrets Distribution**. In this phase, the system authority distributes secrets for devices (according to an existing symmetric key establishment protocol).
- **Session Key Establishment**. In this phase, devices establish secret session keys at higher layers by calling the $KE(\cdot,\cdot)$ protocol. We denote by $KE(\cdot,\cdot)$ a black-box of the employed key establishment protocol.

From the above overview and Figure 5 we can see that there are two types of keys in our model, i.e., the master key (k_i) extracted and shared between the system authority and the i^{th} device D_i during the **Master Key Extraction** phase, and the session key (k_{ij}) established between devices D_i and D_j during the **Session Key Establishment** phase. The following subsection provides the details of our cross-layer key establishment model.

4.2 A Cross-Layer Key Establishment Model

This subsection presents the details of our cross-layer key establishment model.

Initialization. In this phase, system parameters are generated. Specifically, for an input security parameter 1^k, the system authority generates secret values $\mathcal{S}$ and public values $\mathcal{P}$ according to an existing symmetric key establishment

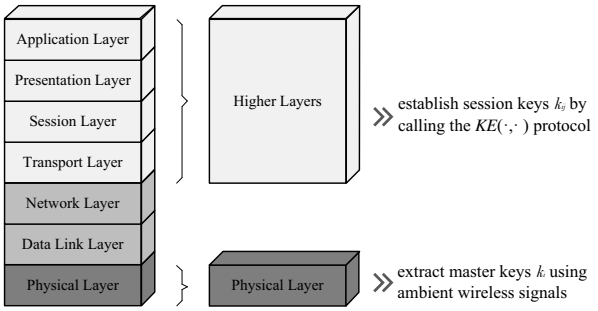

Figure 5: The system model of our design. Specifically, there are two types of keys, i.e., the master key k_i extracted at the physical layer and the session key k_{ij} established at higher layers.

protocol $KE(\cdot,\cdot)$. Then, the system authority chooses a hash function $H(x)$ from a collision-resistant hash family $\mathcal{H}$. The $H(x)$ is used to map arbitrary finite inputs $\{0,1\}^*$ to $\{0,1\}^k$. At the end of this phase, the system authority publishes $H(x)$.

Master Key Extraction. In this phase, the master keys (shared between devices and the system authority) are extracted at the physical layer. We denote by $\mathcal{D}$ the set of N devices in the CPS. For the i^{th} device D_i ($D_i \in \mathcal{D}$ and $i = 1, 2, \ldots, N$), it extracts and obtains a secret master key k_i (shared with the system authority) by running Mathur et al.'s algorithm [19] (as reviewed in Subsection 3.2). At the end of this phase, each device extracts a secret master key shared with the system authority.

Secrets Distribution. In this phase, the system authority distributes secrets for each device. We assume that in the employed symmetric key establishment protocol $KE(\cdot,\cdot)$, device D_i needs to be loaded with m secrets S_js, e.g., $S_1, S_2, \ldots, S_m$. Thus, in this phase, the system authority and the device D_i execute the following operations in order to distribute the secrets S_js for device D_i (Figure 6 shows the main operations):

- The device D_i generates a random number R_i from the field $GF(q)$ (where q has length k bits), and computes $C_{V1} = k_i \oplus R_i$. Here "k_i" is the extracted master key shared between the system authority and the device D_i, and "$\oplus$" is the XOR operations. Completing these operations, the device D_i sends the secrets distribution request $\{\text{req secrets distribution} : C_{V1}, id_i, id_{sys}\}$ to the system authority. Here,"id_i" is the identifier of the device D_i, and "id_{sys}" is the identifier of the system authority.

- Receiving the request, the system authority computes $C_j = H(k_i||j) \oplus S_j$, where $j = 1, 2, \ldots, m$. Here, "$H(x)$" is the public collision-resistant hash function, and "||" is the string concatenation. Then, the system authority generates a random number R_v from the field $GF(q)$, and computes $R_i = C_{V1} \oplus k_i$, $C_{V2} = H(S_1||S_2||\cdots||S_m) \oplus R_v$, $C_{V3} = H(R_i||m+1) \oplus H(R_v)$. Completing the above operations, the system authority sends the message $V_1 =< id_{sys}, id_i, C_1, C_2, \ldots, C_m, C_{V2}, C_{V3} >$ to the device D_i.

- Receiving the message V_1, the device D_i computes $S_j = C_j \oplus H(k_i||j)$, and obtains the m secrets S_js. Then, the D_i computes $R_v = C_{V2} \oplus H(S_1||S_2||\cdots||S_m)$ and verifies if $H(R_v) = C_{V3} \oplus H(R_i||m+1)$. If the verification succeeds, the device D_i accepts the m secrets S_js, computes $C_{V4} = H(R_v||S_1||S_2||\cdots||S_m)$, and sends $V_2 =< id_i, id_{sys}, C_{V4} >$ to the system authority. Otherwise, the device D_i outputs the undefined symbol "$\perp$" and terminates the communications immediately.

- Receiving the message V_2, the system authority computes $H(R_v||S_1||S_2||\cdots||S_m)$ and verifies if $H(R_v||S_1||S_2||\cdots||S_m) = C_{V4}$. If the verification succeeds, the system authority accepts the device D_i as a legitimate device. Otherwise, the system authority outputs the undefined symbol "$\perp$" and terminates the communications immediately.

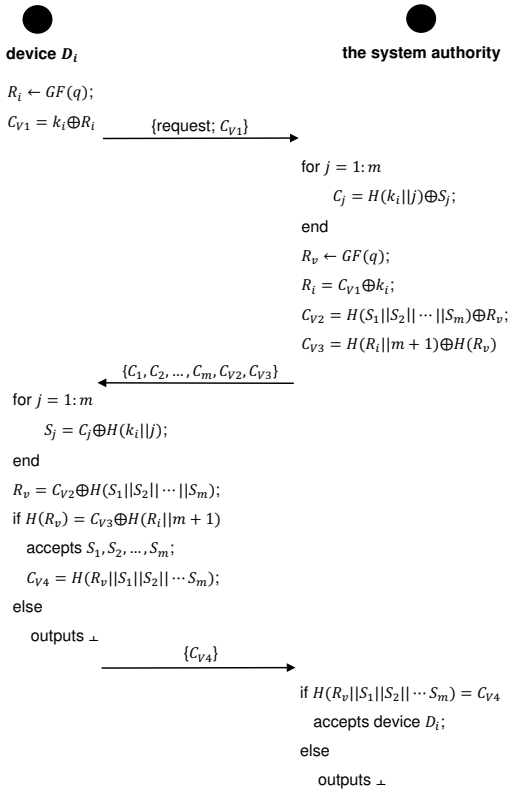

Figure 6: Operations in the Secrets Distribution phase of the model.

Completing the **Secrets Distribution** phase, each device is distributed with m secrets S_js.

Session Key Establishment. The i^{th} and j^{th} devices can establish a secret session key by calling the employed key establishment protocol $KE(\cdot,\cdot)$. Recall that the system authority distributes each device with m secrets (according to the employed key establishment protocol) in the **Secrets Distribution** phase, thus, the i^{th} and j^{th} devices can establish a secret session key by calling the key establishment protocol (i.e., calling $KE(id_i, id_j)$).

This completes the description of our cross-layer key establishment model. To facilitate understanding, in Section 6,

we implement the proposed model and convert existing symmetric key establishment protocols into cross-layer key establishment protocols such that the protocols can be directly implemented in the CPS.

5. SECURITY ANALYSIS

This section analyses the security of our cross-layer key establishment model.

Theorem. *Assuming that secret master keys can be extracted at the physical layer, the employed key establishment protocol is a secure key establishment protocol (in the reviewed security model in Subection 3.1), and $H(x)$ is a collision-resistant hash function, then the proposed cross-layer key establishment model is a secure key establishment model.*

Before proving the above Theorem, we briefly introduce the logic of our security proof. Let Exp_0 be the experiment in which the adversary $\mathcal{A}$ attacks the proposed model. Then, a sequence of experiments are introduced. In order to facilitate analysis, a simulator is employed to interact with the adversary. Specifically, when the adversary queries, the simulator executes the appropriate algorithm and makes a response. Under the assumptions that secret master keys can be extracted at the physical layer, the employed key establishment protocol is a secure key establishment protocol (in the reviewed security model in Subection 3.1), and $H(x)$ is a collision-resistant hash function, experiments Exp_1 to Exp_5 prove that the adversary has the probability

$$P^{compromise}_{AKE,\mathcal{A},P}(k) \leq \varepsilon(k)$$

to compute and obtain the secret session key established between two benign devices. Here, $\varepsilon(k)$ is a negligible probability. Now, details of security proof are given in the following paragraphs.

Proof. Let Exp_0 is the experiment, in which an adversary $\mathcal{A}$ attacks the proposed model. Thus, $P^{compromise}_{AKE,\mathcal{A},Exp_0}(k) = P^{compromise}_{AKE,\mathcal{A},P}(k)$.

Experiment Exp_1: In experiment Exp_1, the adversary obtains the eavesdropped messages. As we assumed that the adversary can eavesdrop the communications. Thus, in experiment Exp_1, the simulator outputs the following messages:

$$\begin{aligned} &C_{V1} = k_i \oplus R_i \\ &C_j = H(k_i||j) \oplus S_j, \text{ where } 1 \leq j \leq m, \\ &C_{V2} = H(S_1||S_2||\cdots||S_m) \oplus R_v, \\ &C_{V3} = H(R_i||m+1) \oplus H(R_v), \\ &C_{V4} = H(R_v||S_1||S_2||\cdots||S_m). \end{aligned} \tag{1}$$

Recall that k_i is the extracted master key, $S_1, S_2, \ldots, S_m$ are secrets generated by the system authority, R_i and R_v are random numbers generated by device D_i and the system authority. Assuming that secret master keys can be extracted at the physical layer by running the key extract algorithm [19], then, the master key k_i is a secret key shared between the device D_i and the system authority. Additionally, we assume that the $H(x)$ is a collision-resistant hash function, and it is used to map $\{0,1\}^*$ to $\{0,1\}^k$. Thus, C_{V1}, C_js, C_{V2}, and C_{V3} are one-time pads. Recall that the one-time pad is a perfectly secure cipher [24, 11], thus, the adversary has probabilities $\frac{1}{2^k}$ and $\frac{1}{2^{m \cdot k}}$ to correctly compute k_i and S_js (i.e., randomly guess) by making use of the eavesdropped equation set 1.

Besides, we assume that the employed key establishment protocol is a secure key establishment protocol in the reviewed security model (as shown in Subsection 3.1). Namely, the adversary has a negligible probability ε_0 to compute and obtain the session keys established between non-compromised devices by making use of the eavesdropped messages (the messages transmitted during the calling of the employed key establishment protocol). Thus, we have $|P^{compromise}_{AKE,\mathcal{A},Exp_1}(k) - P^{compromise}_{AKE,\mathcal{A},Exp_0}(k)| \leq \varepsilon_1 = Q_1(k) \cdot \varepsilon_0 + \frac{Q_1(k)}{2^k}$, where $Q_1(k)$ is the maximal number of executing experiment Exp_1 executed by the adversary.

Experiment Exp_2: In experiment Exp_2, the adversary queries {req secrets distribution: c_{v1}, id_i, id_{sys}}. Receiving the query, the simulator generates random numbers k_s, S'_js, R_v from the field $GF(q)$, sets $k_i = k_s$, computes $C'_j = H(k_i||j) \oplus S'_j$, $R'_i = c_{v1} \oplus k_i$, $C_{V2} = H(S'_1||S'_2||\cdots S'_m) \oplus R_v$, $C_{V3} = H(R'_i||m+1) \oplus H(R'_v)$. Then, the simulator sends the messages $V'_1 =< id_{sys}, id_i, C'_1, C'_2, \ldots, C'_m, C_{V2}, C_{V3} >$ to the adversary. The remainder operations are the same as in Exp_1. Assuming that secret master keys can be extracted at the physical layer by running the key extraction algorithm [19], and $H(x)$ is a collision-resistant hash function, the transmitted messages C'_js, C_{V2} and C_{V3} are one-time pads. Namely, the adversary has probabilities $\frac{1}{2^k}$ and $\frac{1}{2^{m \cdot k}}$ to correctly compute and obtain k_i and S'_js by making use of the received messages V'_1. Thus, we have $|P^{compromise}_{AKE,\mathcal{A},Exp_2}(k) - P^{compromise}_{AKE,\mathcal{A},Exp_1}(k)| \leq \varepsilon_2 = \frac{Q_2(k)}{2^k}$, where $Q_2(k)$ is the maximal number of querying {req secrets distribution: c_{v1}, id_i, id_{sys}} executed by the adversary.

Experiment Exp_3: In experiment Exp_3, the adversary queries $v_1 =< id_{sys}, id_i, c_1, c_2, \ldots, c_m, c_{v2}, c_{v3} >$. Receiving the query, the simulator generates random numbers K, R_s from the field $GF(q)$ and sets $k_i = K$. Then the simulator computes $S'_j = c_j \oplus H(k_i||j)$, $R'_v = c_{v2} \oplus H(S'_1||S'_2||\cdots S'_m)$, and sets $C'_{V4} = R_s$. Completing these operations, the simulator sends $V'_2 =< id_i, id_{sys}, C'_{V4} >$ to the adversary. The remainder operations are the same as in Exp_2. As long as secret master keys can be extracted at the physical layer, and $H(x)$ is a collision-resistant hash function, the adversary cannot compute and find the difference between Exp_3 and Exp_2. Thus, we have $P^{compromise}_{AKE,\mathcal{A},Exp_3}(k) = P^{compromise}_{AKE,\mathcal{A},Exp_2}(k)$.

Experiment Exp_4: In experiment Exp_4, the adversary queries $< id_i, id_{sys}, c_{v4} >$. Receiving the query, the simulator directly outputs the undefined symbol "$\perp$" and terminates the communication immediately. The remainder operations are the same as in Exp_3. Under the assumptions that secret master keys can be extracted at the physical layer, and $H(x)$ is a collision-resistant hash function, the adversary cannot compute and find the difference between Exp_4 and Exp_3. Thus, we have $P^{compromise}_{AKE,\mathcal{A},Exp_4}(k) = P^{compromise}_{AKE,\mathcal{A},Exp_3}(k)$.

Experiment Exp_5: In experiment Exp_5, the adversary queries {req : $KE(id_i, id_j)$}. Receiving the query, the simulator runs the employed key establishment protocol $KE(\cdot,\cdot)$ and outputs the simulator generated messages to the adversary. Recall that calling a certain employed key establishment protocol, some public messages may be transmitted. Thus, the simulator generates random messages and outputs the simulator generated messages when running the employed key establishment protocol. The remainder operations are the same as in Exp_4. We assume that the employed key establishment protocol is a secure key establishment protocol in the reviewed security model (introduced

in Subsection 3.1). Namely, the adversary has a negligible probability ε_0' to compute and obtain the session keys established between noncompromised devices when it actively attacks the employed key establishment protocol. Thus, we have $|P^{compromise}_{AKE,\mathcal{A},Exp_5}(k) - P^{compromise}_{AKE,\mathcal{A},Exp_4}(k)| \leq \varepsilon_3 = Q_3(k) \cdot \varepsilon_0'$, where $Q_3(k)$ is the maximal number of querying $\{\text{req} : KE(id_i, id_j)\}$ executed by the adversary.

The above analysis shows that

$$|P^{compromise}_{AKE,\mathcal{A},Exp_5}(k) - P^{compromise}_{AKE,\mathcal{A},P}(k)| \leq \varepsilon(k), \qquad (2)$$

where $\varepsilon(k) = \varepsilon_1(k) + \varepsilon_2(k) + \varepsilon_3(k)$ is a negligible probability. Equation 2 illustrates that, under the assumptions: i). secret master keys can be extracted at the physical layer; ii). the employed key establishment protocol is a secure key establishment protocol (in the reviewed security model in Subsection 3.1); and iii). $H(x)$ is a collision-resistant hash function, the proposed cross-layer key establishment model is a secure key establishment model. This completes the proof of the theorem. □

6. PERFORMANCE ANALYSIS

Subsection 4.2 presents our cross-layer key establishment model. Recall that implementing the proposed model, existing symmetric key establishment protocols can be converted into cross-layer key establishment protocols such that they can be directly implemented in scenarios, such as the CPS. Thus, in this section, we analyse the performance of our model by showing several examples.

6.1 Converting the Key Pre-Distribution Protocol [7]

Motivated by the observations that the secrets sharing assumption can be weakened by implementing our model, we remove the assumption and convert the q-KP protocol [7] into a cross-layer key establishment protocol. The detailed operations are as follows:

Initialization. In this phase, random keys are generated (according to the employed q-KP protocol [7]). For an input security parameter 1^k, the system authority chooses parameter Z, generates a set of random keys $\mathcal{K} = \{K_1, K_2, \ldots, K_Z\}$ and the key identifiers id_is. Then, the system authority chooses a hash function $H(x)$ from a collision-resistant hash family $\mathcal{H}$. At the end of this phase, the system authority publishes $H(x)$.

Master Key Extraction. In this phase, sensor nodes extract master keys (shared with the system authority) at the physical layer. We denote by $\mathcal{D}$ the set of N nodes, i.e., $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$. For the i^{th} node D_i, it extracts a secret master key k_i (shared with the system authority) by executing the operations presented in the **Master Key Extraction** phase of our model. At the end of this phase, each sensor node obtains a secret master key shared with the system authority.

Secrets Distribution. In this phase, the system authority distributes keys for sensor nodes. Specifically, for each node, system authority randomly chooses m keys from $\mathcal{K}$, and distributes the m keys by executing the operations presented in the **Secrets Distribution** phase of our model (please refer to Subsection 4.2 for details). At the end of this phase, each sensor node obtains m randomly chosen keys.

Session Key Establishment. In this phase, the i^{th} and j^{th} nodes can establish a session key by calling the employed key establishment protocol (i.e., the q-KP protocol [7]). For instance, calling q-KP(id_i, id_j), the i^{th} and j^{th} nodes broadcast the identifiers of distributed keys (obtained in the *Secrets Distribution* phase). We assume that the i^{th} and j^{th} nodes share q' keys. Thus, according to the q-KP protocol [7], the i^{th} and j^{th} nodes can establish a session key $k_{ij} = H(K_1||K_2||\cdots||K_{q'})$ when $q' \geq q$.

This completes the description of converting the classical key pre-distribution protocol (i.e., the q-KP protocol [7]) into a cross-layer key establishment protocol by implementing our proposed model.

6.2 Converting the Polynomial-Based Key Establishment Protocol [17]

Motivated by the observations that the secrets sharing assumption can be weakened by implementing our model, we remove the assumption and convert the PKE protocol [17] into a cross-layer key establishment protocol. The detailed operations are as follows:

Initialization. In this phase, polynomials are generated (according to the employed PKE protocol). For an input security parameter 1^k, the system authority generates a set of bivariate t-degree polynomials $\mathcal{F}$ over the finite field $GF(q)$, where q has length of k bits. We denote by ID_i the identifier of the i^{th} polynomial $f_i(x,y) = \sum_{i,j=0}^{t} a_{ij}x^i y^j$, and $f_i(x,y) \in \mathcal{F}$. Then, the system authority chooses a hash function $H(x)$ from a collision-resistant hash family $\mathcal{H}$. At the end of this phase, the system authority publishes $H(x)$.

Master Key Extraction. In this phase, sensor nodes extract master keys (shared with the system authority) at the physical layer. We denote by $\mathcal{D}$ the set of N nodes, i.e., $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$. For the i^{th} node D_i, it extracts a secret master key k_i (shared with the system authority) by executing the operations presented in the **Master Key Extraction** phase of our model. At the end of this phase, each sensor node obtains a secret master key shared with the system authority.

Secrets Distribution. In this phase, the system authority distributes the shares of polynomials for sensor nodes. Specifically, for the i^{th} node, system authority randomly chooses a subset of polynomials $\mathcal{F}_i$ from the polynomial pool $\mathcal{F}$, and computes the shares of the chosen polynomials. Then, the system authority distributes the shares to the i^{th} node by executing the operations presented in the **Secrets Distribution** phase of our model (please refer to Subsection 4.2 for details). At the end of this phase, each sensor node obtains the shares of a subset of polynomials $\mathcal{F}_i$.

Session Key Establishment. In this phase, the i^{th} and j^{th} nodes can establish a session key by calling the employed key establishment protocol (i.e., the PKE protocol). For instance, calling $PKE(id_i, id_j)$, the i^{th} and j^{th} nodes broadcast the identifiers of distributed polynomials ID_is (obtained in the *Secrets Distribution* phase). Then, according to the PKE protocol, the i^{th} and j^{th} nodes can establish a session key using the shared polynomial(s).

This completes the description of converting the polynomial-based key establishment protocol (i.e., the PKE protocol [17]) into a cross-layer key establishment protocol by implementing our proposed model.

6.3 Converting the Matrix-Based Key Establishment Protocol [9]

Motivated by the observations that the secrets sharing

assumption can be weakened by implementing our model, we remove the assumption and convert the MKE protocol [9] into a cross-layer key establishment protocol. The detailed operations are as follows:

Initialization. In this phase, secret and public matrices are generated (according to the employed MKE protocol [9]). For an input security parameter 1^k, the system authority: 1. chooses system parameter λ, and designs a $(\lambda+1)\times N$ matrix G over a finite field $GF(q)$ (where q has length k bits)

$$G = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ s & s^2 & s^3 & \dots & s^n \\ s^2 & (s^2)^2 & (s^3)^2 & \dots & (s^n)^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s^\lambda & (s^2)^\lambda & (s^3)^\lambda & \dots & (s^n)^\lambda \end{pmatrix}.$$

Here, "N" is the number of nodes in the networks; 2. designs ω secret symmetric $(\lambda+1)\times(\lambda+1)$ matrices $D_1, D_2, \dots, D_\omega$ in $GF(q)$, and computes matrices $A_1 = (D_1 \cdot G)^T, A_2 = (D_2 \cdot G)^T, \dots, A_\omega = (D_\omega \cdot G)^T$. Here, "$\cdot$" is the matrix dot product, and "T" is the matrix transpose; 3. chooses a hash function $H(x)$ from a collision-resistant hash family $\mathcal{H}$. The $H(x)$ is used to map arbitrary finite inputs $\{0,1\}^*$ to members of the field $GF(q)$. At the end of this phase, the system authority publishes $H(x)$.

Master Key Extraction. In this phase, sensor nodes extract master keys (shared with the system authority) at the physical layer. We denote by $\mathcal{D}$ the set of N nodes, i.e., $\mathcal{D} = \{D_1, D_2, \dots, D_N\}$. For the i^{th} node D_i, it extracts a secret master key k_i (shared with the system authority) by executing the operations presented in the **Master Key Extraction** phase of our model. At the end of this phase, each sensor node obtains a secret master key shared with the system authority.

Secrets Distribution. In this phase, the system authority distributes secrets for sensor nodes. For instance, the system authority randomly selects τ A_is, distributes the k^{th} row of each selected A_i and the k^{th} key seed s^k of G (the k^{th} key seed is the second element in the k^{th} column of matrix G) for the k^{th} node. The distribution can be completed by executing the operations presented in the **Secrets Distribution** phase of our model (please refer to Subsection 4.2 for details). At the end of this phase, each sensor node obtains τ rows of matrices A_is and a key seed.

Session Key Establishment. In this phase, the i^{th} and j^{th} nodes can establish a session key by calling the employed key establishment protocol (i.e., the MKE protocol). For instance, calling $MKE(id_i, id_j)$, the i^{th} and j^{th} nodes broadcast the identifiers of distributed matrices. Then, two nodes can establish a session key when they are loaded with rows from the same matrices A_is.

This completes the description of converting the matrix-based key establishment protocol (i.e., the MKE protocol [9]) into a cross-layer key establishment protocol by implementing our proposed model.

Due to the limitation of space, we only provide the above three examples to convert symmetric key establishment protocols [7, 17, 9] into cross-layer key establishment protocols. Recall that in existing symmetric key establishment protocols, it is assumed that devices are loaded with secrets via secure channels or when the system authority is off-line. In certain applications, such as in the CPS, wireless devices are produced by different companies. It is not practical to assume that the devices are pre-loaded with certain secrets when they leave companies. As a result, the existing symmetric key establishment protocols cannot be directly implemented in these applications. Motivated by this observation, this paper present a cross-layer key establishment model such that existing symmetric key establishment protocols can be directly implemented in the CPS by employing the proposed model.

Our cross-layer key establishment model can convert existing symmetric key establishment protocols into cross-layer key establishment protocols such that these protocols can be directly implemented in the CPS. It is achieved due to the reason that in our model, wireless devices extract and obtain secret master keys k_is (shared with the system authority) by running Mathur et al.'s algorithm [19] (as reviewed in Subsection 3.2). Making use of the extracted master keys, "a secure channel" can be established between the system authority and wireless devices. Recall that in existing symmetric key establishment protocols, it is assumed that devices are loaded with certain secrets via secure channels. Thus, implementing our model, the secrets sharing assumption in existing symmetric key establishment protocols can be removed.

From the above analysis we can see that implementing the proposed model, existing symmetric key establishment protocols can be directly employed in the scenarios when devices do not pre-share any secrets. However, it introduces extra energy consumptions. The reason is that in our model, devices need to extract master keys by running the key extraction algorithm [19]. In [19], a linear error correcting code is used. Thus, for an n-bit master key, the extra computational complexity is $O(n)$. As analysed in [19] that a number of factors (such as the distance between a device and the system authority, the wavelength of the public source, whether the devices are held stationary or moved, and the number of RF sources being monitored) affect the performance of the key extraction algorithm. For instance, when a device and the system authority are $d = 0.05\lambda$ apart and they use 10 sources in parallel, it takes around 33 seconds (from the TV signals) and 102.5 seconds (from the FM signals) to extract a 128-bit master key (when both the device and the system authority are stationary). It takes around 10.2 seconds (from the TV signals) and 41.2 seconds (from the FM signals) to extract a 128-bit master key, when both the device and the system authority moved slowly.

7. CONCLUSION

To secure the communications, secret session keys need to be established between wireless devices. In existing symmetric key establishment protocols, it is assumed that devices are pre-loaded with secrets. In practice, however, wireless devices in the CPS are produced by different companies. Thus, it is not practical to assume that the devices are pre-loaded with certain secrets when they leave companies. As a result, existing symmetric key establishment protocols cannot be directly implemented in the CPS. Moreover, it is impractical to extract session keys using ambient wireless signals when a large number of session keys need to be established. However, it should be a reasonable idea to alleviate these problems by utilising the characteristics of these two types of key establishment protocols cooperative-

ly. Motivated by these observations, this paper presents a cross-layer key establishment model for wireless devices in the CPS. Specifically, implementing our model, each device only extracts a master key (shared with the system authority) at the physical layer using the ambient wireless signals. Making use of the extracted master keys, the system authority distributes secrets for devices (according to the employed symmetric key establishment protocol). Completing these operations, devices can establish session keys at higher layers by calling the employed key establishment protocol. Additionally, we prove the security of the proposed model and analyse the performance of it by implementing the proposed model. The analysis shows that existing symmetric key establishment protocols can be directly implemented in the CPS by employing the new model.

Acknowledgements

Xinyi Huang is supported by Distinguished Young Scholars Fund of Fujian (2016J06013).

8. REFERENCES

[1] IEEE standard for local and metropolitan area networks–part 15.4: low-rate wireless personal area networks (LR-WPANs). *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)*, pages 1–314, Sept 2011.

[2] S. Ali, S. B. Qaisar, H. Saeed, M. F. Khan, M. Naeem, and A. Anpalagan. Network challenges for cyber physical systems with tiny wireless devices: a case study on reliable pipeline condition monitoring. *Sensors*, 15(4):7172–7205, 2015.

[3] P. Bahl and V. N. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *Proceedings IEEE INFOCOM 2000, The Conference on Computer Communications, Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Reaching the Promised Land of Communications, Tel Aviv, Israel, March 26-30, 2000*, pages 775–784. IEEE, 2000.

[4] R. Blom. An optimal class of symmetric key generation systems. In T. Beth, N. Cot, and I. Ingemarsson, editors, *Advances in Cryptology: Proceedings of EUROCRYPT 84, A Workshop on the Theory and Application of of Cryptographic Techniques, Paris, France, April 9-11, 1984, Proceedings*, volume 209 of *Lecture Notes in Computer Science*, pages 335–338. Springer, 1984.

[5] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung. Perfectly-secure key distribution for dynamic conferences. In E. F. Brickell, editor, *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, volume 740 of *Lecture Notes in Computer Science*, pages 471–486. Springer, 1992.

[6] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry. Challenges for securing cyber physical systems. In *Workshop on future directions in cyber-physical systems security*, page 5, 2009.

[7] H. Chan, A. Perrig, and D. X. Song. Random key predistribution schemes for sensor networks. In *2003 IEEE Symposium on Security and Privacy (S&P 2003), 11-14 May 2003, Berkeley, CA, USA*, page 197. IEEE Computer Society, 2003.

[8] J. Delvaux, R. Peeters, D. Gu, and I. Verbauwhede. A survey on lightweight entity authentication with strong PUFs. *ACM Computing Surveys*, 48(2):26, 2015.

[9] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili. A pairwise key predistribution scheme for wireless sensor networks. *ACM Transactions on Information and System Security*, 8(2):228–258, 2005.

[10] L. Eschenauer and V. D. Gligor. A key-management scheme for distributed sensor networks. In V. Atluri, editor, *Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002*, pages 41–47. ACM, 2002.

[11] X. He and A. Yener. The role of feedback in two-way secure communications. *IEEE Transactions on Information Theory*, 59(12):8115–8130, 2013.

[12] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. On the effectiveness of secret key extraction from wireless signal strength in real environments. In K. G. Shin, Y. Zhang, R. Bagrodia, and R. Govindan, editors, *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking, MOBICOM 2009, Beijing, China, September 20-25, 2009*, pages 321–332. ACM, 2009.

[13] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: attacks and countermeasures. *Ad Hoc Networks*, 1(2-3):293–315, 2003.

[14] S. Khan, N. A. Alrajeh, and K.-K. Loo. Secure route selection in wireless mesh networks. *Computer Networks*, 56(2):491–503, 2012.

[15] A. P. Lauf, R. A. Peters, and W. H. Robinson. A distributed intrusion detection system for resource-constrained devices in ad-hoc networks. *Ad Hoc Networks*, 8(3):253–266, 2010.

[16] E. A. Lee. Cyber physical systems: design challenges. In *11th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC 2008), 5-7 May 2008, Orlando, Florida, USA*, pages 363–369. IEEE Computer Society, 2008.

[17] D. Liu, P. Ning, and R. Li. Establishing pairwise keys in distributed sensor networks. *ACM Transactions on Information and System Security*, 8(1):41–77, 2005.

[18] Y. Liu and J. Li. Key predistribution based broadcast authentication scheme for wireless sensor networks. In *Fourth International Conference on Frontier of Computer Science and Technology, FCST 2009, Shanghai, China, 17-19 December, 2009*. IEEE Computer Society, 2009.

[19] S. Mathur, R. D. Miller, A. Varshavsky, W. Trappe, and N. B. Mandayam. ProxiMate: proximity-based secure pairing using ambient wireless signals. In A. K. Agrawala, M. D. Corner, and D. Wetherall, editors, *Proceedings of the 9th International Conference on Mobile Systems, Applications, and*

Services (MobiSys 2011), Bethesda, MD, USA, June 28 - July 01, 2011, pages 211–224. ACM, 2011.

[20] S. Mathur, W. Trappe, N. B. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In J. J. Garcia-Luna-Aceves, R. Sivakumar, and P. Steenkiste, editors, *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking, MOBICOM 2008, San Francisco, California, USA, September 14-19, 2008*, pages 128–139. ACM, 2008.

[21] C. Perera, R. Ranjan, L. Wang, S. U. Khan, and A. Y. Zomaya. Big data privacy in the internet of things era. *IT Professional*, 17(3):32–39, 2015.

[22] C. Pham. Scheduling randomly-deployed heterogeneous video sensor nodes for reduced intrusion detection time. In M. K. Aguilera, H. Yu, N. H. Vaidya, V. Srinivasan, and R. R. Choudhury, editors, *ICDCN*, volume 6522 of *Lecture Notes in Computer Science*, pages 303–314. Springer, 2011.

[23] S. N. Premnath, S. Jana, J. Croft, P. L. Gowda, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. Secret key extraction from wireless signal strength in real environments. *IEEE Transaction on Mobile Computing*, 12(5):917–930, 2013.

[24] C. E. Shannon. Communication theory of secrecy systems. *Bell system technical journal*, 28(4):656–715, 1949.

[25] Q. Wang, H. Su, K. Ren, and K. Kim. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In *INFOCOM 2011. 30th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 10-15 April 2011, Shanghai, China*, pages 1422–1430. IEEE, 2011.

[26] Q. Wang, K. Xu, and K. Ren. Cooperative secret key generation from phase estimation in narrowband fading channels. *IEEE Journal on Selected Areas in Communications*, 30(9):1666–1674, 2012.

[27] M. Yampolskiy, P. Horváth, X. D. Koutsoukos, Y. Xue, and J. Sztipanovits. A language for describing attacks on cyber-physical systems. *International Journal of Critical Infrastructure Protection*, 8:40–52, 2015.

[28] Z. Yang and Y. Liu. Understanding node localizability of wireless ad hoc and sensor networks. *IEEE Transactions on Mobile Computing*, 11(8):1249–1260, 2012.

[29] B. Zan, M. Gruteser, and F. Hu. Key agreement algorithms for vehicular communication networks based on reciprocity and diversity theorems. *IEEE Transactions on Vehicular Technology*, 62(8):4020–4027, 2013.

[30] K. Zeng. Physical layer key generation in wireless networks: challenges and opportunities. *IEEE Communications Magazine*, 53(6):33–39, 2015.

[31] K. Zeng, D. Wu, A. J. Chan, and P. Mohapatra. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. In *INFOCOM 2010. 29th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 15-19 March 2010, San Diego, CA, USA*, pages 1837–1845. IEEE, 2010.