Physical Layer Key Generation in Wireless Networks: Challenges and Opportunities

Kai Zeng

ABSTRACT

Physical layer key generation that exploits reciprocity and randomness of wireless fading channels has attracted considerable research attention in recent years. Although theoretical study has shown its potential to generate information-theoretic secure keys, great challenges remain when transforming the theory into practice. This article provides an overview of the physical layer key generation process and discusses its practical challenges. Different passive and active attacks are analyzed and evaluated through numerical study. A new key generation scheme using random probing signals, and combining user generated randomness and channel randomness, is introduced as a countermeasure against active attacks. The numerical results show that the proposed scheme achieves higher security strength than existing schemes using constant probing signals under active attacks. Future research topics on physical layer key generation are discussed.

INTRODUCTION

With the proliferation of the Internet of Things (IoT), diversified wireless devices need to establish secure communications on the fly. One common way to secure the communication between wireless devices is to generate a symmetric key between them and use it to encrypt/decrypt the message. One conventional mechanism to generate a shared secret key between two parties is the Diffie-Hellman (D-H) key exchange protocol. However, the computation overhead of D-H protocol is significant due to expensive exponential operation, which is undesirable for resource constrained devices such as embedded sensors, wearable devices, RFIDs, and so on. Furthermore, with the ever increasing computing power of attackers, D-H protocol has to increase the key length in order to maintain a certain level of security strength, which in turn aggravates the computation overhead.

An alternative way to generate a shared secret key between wireless devices is to exploit the reciprocity of the random fading channel [1–5]. This mechanism is generally called physical layer key generation, in which wireless devices measure highly correlated wireless channel characteristics (e.g., channel impulse responses or received signal strengths) and use them as shared random sources to generate a shared key. In theory, in a rich multipath scattering environment, a passive attacker who is more than a half-wavelength away from the legitimate users will obtain uncorrelated channel measurements, and thus cannot infer much information about the generated key. The physical layer key generation mechanisms do not require expensive computation and have the potential to achieve information-theoretic security, in the sense that the secrecy of the generated key is not dependent on the hardness of a computational problem but relies on the physical laws of the wireless fading channels.

Due to its attractive features of lightweight and information-theoretic security, physical layer key generation has gained considerable attention in recent years. A typical key generation process includes channel probing, randomness extraction, quantization, reconciliation, and privacy amplification. Although theoretical study provides guideline on designing physical layer key agreement protocols, there are still significant challenges remaining to achieve an efficient and secure-proven key generation scheme in practice. The major challenges lie in the difficulty of measuring the information leaked to eavesdroppers, tackling channel measurement correlations, reducing reconciliation overhead, and deciding on the compression ratio in the privacy amplification stage.

In this article, we give an overview of the general key generation process, and discuss its practical challenges and possible solutions under passive attacks in the following section. The active attacks against physical layer key generation are summarized after that, and a new key generation scheme using random probing signals and combining user generated randomness and channel randomness is presented. The security strength of the proposed key generation scheme

The author is with George Mason University.

0163-6804/15/$25.00 © 2015 IEEE

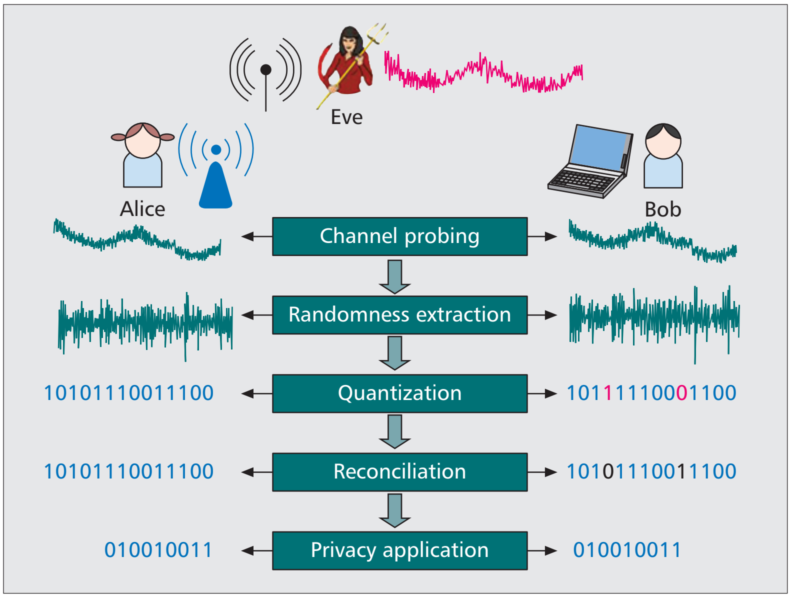

Figure 1. Secret key generation model and steps under passive attack.

and the traditional one using constant probing signals are analyzed and compared under both passive and various active attacks. We then discuss future research directions of physical layer key generation. Conclusions are given in the final section.

Physical Layer Key Generation under Passive Attacks

We first introduce the key generation process, and discuss the practical challenges and possible solutions under passive attacks. We illustrate an application scenario in Fig. 1 where two legitimate users, Alice (an access point) and Bob (a laptop), aim to generate a shared secret key using channel measurements. There is a passive attacker, Eve, who can overhear all the transmissions from Alice and Bob.

Key Generation Primitives

Alice and Bob generally apply the following five steps, illustrated in Fig. 1, to generate a key: channel probing, randomness extraction, quantization, information reconciliation, and privacy amplification [2, 6].

Channel Probing: This is used to collect channel measurements by Alice and Bob. The channel measurements can be channel state information (CSI), received signal strength (RSS), or phase. In this step, Alice and Bob exchange channel probing signals with each other. One channel probing contains a pair of bidirectional channel probing with a short lag of time assuming a half-duplex radio. The received signals are usually modeled as the transmitted sounding signal timing (in the frequency domain) channel gain plus noise. Alice and Bob observe highly correlated received signals due to channel reciprocity.

Randomness Extraction: The received signals at Alice and Bob may contain deterministic parts that can be determined or inferred by the attacker. For example, in Fig. 1, the received signals at Alice and Bob have the same fluctuation pattern on a large scale. This fluctuation is determined by the distance between Alice and Bob. If Eve is close to one of them, she will also observe this large-scale change. Therefore, Alice and Bob should not use this large-scale component to generate shared keys. Otherwise, the key will be easily determined by the attacker. Alice and Bob need to extract randomness caused by channel fading to generate shared keys by removing the large-scale component. A moving window average method can be used to extract the small-scale randomness [6].

Quantization: This is used to quantize the extracted random channel measurements into bits.

Information Reconciliation: This is a form of error correction carried out between Alice and Bob in order to ensure that the keys generated separately on both sides are identical. Due to imperfect reciprocity, the extracted bits at Alice and Bob sides after quantization are usually not identical, although they may be highly similar. This imperfection mainly comes from the fact that Alice and Bob cannot measure the channel at the same time due to the half-duplex property of the radio. Furthermore, the noises at Alice's and Bob's sides are usually independent. During reconciliation, parity bit information may be exchanged to correct errors, and a certain amount of bit information will be revealed to Eve.

Privacy Amplification: This is a method for eliminating Eve's partial information about the key and the correlation among the bits. Eve's partial information comes from eavesdropping during both the probing and reconciliation phases.

Although there has already been intensive study of physical layer key generation under passive attacks [2–4, 6, 7], significant challenges and open issues remain to design an efficient and security-proven key generation scheme in practice. In the following subsections, we discuss these practical challenges and possible solutions.

Difficulty in Estimating Leaked Information in Practice

The secret key capacity is defined as the conditional mutual information between Alice and Bob given Eve's observation [7]. In theory, we can compute various bounds of key capacity given the assumption of knowing eavesdropping CSI. However, in practice, it is very hard to estimate how much information is leaked to a passive eavesdropper. Experimental work has demonstrated that there is a strong correlation in measurements observed by passive eavesdroppers located significantly greater than a half-wavelength away from legitimate devices [8]. It may be due to a poor multipath scattering environment or interference. Therefore, there is not a clear safeguard distance to ensure the secrecy of the generated key. Furthermore, it is hard to know the locations or number of passive eavesdroppers in practice, which introduces difficulty in estimating the leaked information.

This is probably the most critical open issue hindering the design of a security-proven physical layer key generation scheme in practice. More investigation is needed to address this issue.

One possible solution can be adding noise into the channel to jam the eavesdroppers, creating ambient erasure channels to the eavesdropper [9]. How to guarantee a predictable limit amount of information leaked to eavesdroppers deserves further study.

Reconciliation Overhead

The reconciliation overhead can be significant if the bit agreement ratio is low before the reconciliation. The cause of bit disagreement comes from the imperfect reciprocity of the measured channel characteristics, which is mainly caused by the time lag between the bidirectional channel measurements in the channel probing phase. The reconciliation process is essentially an error correction process with information exchange between Alice and Bob. Existing approaches include the Cascade algorithm and low density parity check (LDPC). It has been shown that if Cascade is used to reconcile two bitstrings having a 10 percent bit mismatch, the number of exposed bits can be around 60 percent.

The reconciliation overhead can be significant if the bit agreement ratio is low before the reconciliation. The cause of bit disagreement comes from the imperfect reciprocity of the measured channel characteristics, which is mainly caused by the time lag between the bidirectional channel measurements in the channel probing phase.

In order to minimize the reconciliation overhead, it is very important to achieve a high bit agreement ratio before reconciliation. One straightforward way to achieve a high bit agreement ratio is to reduce the time lag between bidirectional channel probings. However, this time lag is constrained by the antenna tx/rx turnaround time and the medium access control protocol. Giving higher access priority to the probing frames can be helpful.

A second way to reduce the bit disagreement is to use lower-level quantization at the cost of reduced key length. It has been found that lower-level quantization is more robust against noise, but the key length is reduced significantly [6].

A third way to reduce the reconciliation cost is to preprocess the measured data. A low-pass Savitzky-Golay filter is applied to process the measured RSS indicator (RSSI) traces to reduce the maximum frequency of changes in received signal power arising from motion in a small-scale fading environment [3]. A fractional interpolation filtering mechanism is also proposed to enhance the bit agreement ratio [5]. A Farrow filter is used to estimate the channel measurement at simultaneous instants halfway between the original non-simultaneous measurements.

Spatial and Temporal Correlation

In practice, there are always spatial and temporal correlation between the channel measurements, which will lead to correlated bits in the generated key. Before quantization, we need to de-correlate the channel measurements. Existing solutions include applying discrete Karhunen-Loeve transform (KLT) to convert the measured channel vectors into uncorrelated components [3]. Note that uncorrelated is not the same as independent. Although it holds for Gaussian random vectors, they are not equivalent in general. The KLT guarantees zero covariance between transformed elements, but not higher-order cross-moments. A sophisticated attacker may utilize higher-order cross-moments to predict partial bits in the key.

Privacy Amplification

In the privacy amplification phase, Alice and Bob compress the bitstrings obtained after reconciliation to their real entropy. A universal hash function can be applied. However, in practice, it is nontrivial to decide the compression ratio. As discussed previously, it is very hard to accurately estimate how much information is leaked to eavesdroppers during the channel probing phase. Without knowing the leaked information, it is difficult to decide the compression ratio. Furthermore, due to spatial and temporal correlation of the channel measurements, the generated bits may have inherent correlations, which reduce the entropy of the generated key. In order to estimate the entropy of a bitstring, we usually need a large number of bits, which may not be obtainable in practice. Consider that when generating a 128-bit key, there are 2^{128} possible permutations of the bits. A true random bit generator should generate each of the 2^{128} permutations with equal probability. To estimate the entropy of a finite short bitstring, we may apply the concept of approximate entropy and use the measure of Lempel-Ziv complexity [4].

Physical Layer Key Generation Under Active Attacks

Existing works on physical layer key generation mainly focus on security analysis and protocol design under passive attacks. However, the study of physical layer key agreement techniques under active attacks is largely open.

The existing active attacks can be classified into three categories:

- A disruptive jamming (DJ) attack, which aims to disrupt the key generation process and reduce the key generation rate of legitimate users
- A manipulative jamming (MJ) attack, which injects a signal to manipulate the channel measurements and subsequently compromise a portion of the key
- A channel manipulation (CM) attack, which aims to control the wireless channel between Alice and Bob, thus allowing the attacker to infer the generated key

Disruptive Jamming Attack

Disruptive jamming attacks are proposed against physical layer key generation to disrupt the channel probing process [10]. The attacker injects a jamming signal into the channel in order to minimize the key generation rate between legitimate users.

The injected signals affect the received signals at Alice and Bob, which will introduce inconsistency of the channel measurements, thus reducing the key generation rate. As found in [10], under active attacks, key generation efficiency degrades rapidly with adversarial signal power and external signal interference.

How to achieve a robust secret key generation scheme under active attack is still an open issue. Existing anti-jamming mechanisms such as channel hopping and spread spectrum frequency hopping may be applied to alleviate the jamming effect. It will also be helpful if Alice and Bob can determine or detect which probing frames are contaminated by the jamming signal and discard those contaminated frames or quit the key generation process.

The methodology of integrating user-introduced randomness and channel reciprocity to generate keys is analogous to D-H key exchange. The independent random signals generated at both sides are analogous to the random numbers generated in D-H protocol.

Manipulative Jamming Attack

Instead of disrupting the channel probing process and reduce the key generation rate, a manipulative jamming attacker aims to largely control the channel measurements at legitimate users, thus compromising the generated key.

A manipulative attack is introduced in [11]. To avoid key disagreement, which may lead to attack detection, the attacker waits for injection opportunities when it detects similar RSSs from Alice and Bob, which indicates similar channel gains in the attacking channels. Then the attacker emits a random attacking signal into the channel, which will cause similar RSSs at Alice and Bob. This attack requires the destruction of the legitimate probing frames sent by Alice and Bob by reactive jamming. Under this attack, a portion of received signals at Alice and Bob are controlled by the active attacker. Although Alice and Bob may achieve a high key generation rate, a portion of the shared key bits are compromised without Alice's and Bob's awareness. It is considered a more adverse case than a disruptive jamming attack.

Channel Manipulation Attack

A smart attacker can also try to manipulate and control the channel between Alice and Bob, and largely control the generated key. It was demonstrated in [2] that an adversary can use planned movements in a relative static environment causing desired and predictable changes in the channel between Alice and Bob. For instance, when there is a line-of-sight path between Alice and Bob, an attacker in the middle can block this path to cause an RSS drop at Alice's and Bob's sides. When the attacker moves away, the RSS increases. Then an attacker can randomly block and unblock the line-of-sight path between Alice and Bob to cause a random RSS drop and increase, thus controlling the channel variations and making the generated key predictable.

One possible way to avoid this attack is to run the RSS-measurement-based key extraction scheme only in rich multipath environments where multiple random moving objects are present so that the attacker's movement alone will not be able to change the channel predictably.

Defending against Active Attacks

In this subsection, we present a new key generation scheme to defend against the above mentioned active attacks. The fundamental reason why these active jamming attacks can be successful lies in the fact that Alice and Bob only use the channel measurements to generate the key. If an attacker can manipulate the channel measurements, she can manipulate or infer the generated key. We propose to integrate user-generated randomness into the channel probing, and generate a shared key based on channel reciprocity and combination of user generated randomness and channel randomness.

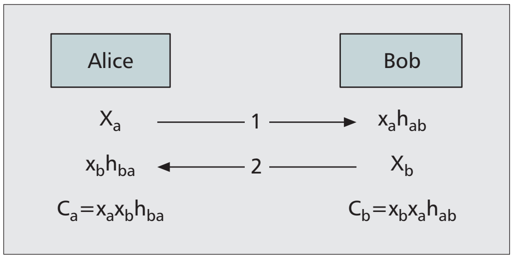

Figure 2. Secret key generation with user-introduced randomness.

The basic idea is illustrated in Fig. 2. When probing the channel, instead of transmitting a constant sounding signal, Alice and Bob transmit independent random signals x_a and x_b. For easy understanding, we ignore the noise in the following explanation.

When there is no attack, Alice and Bob receive $x_b h_{ba}$ and $x_a h_{ab}$, respectively. Note that since x_a and x_b are random signals, neither Alice nor Bob can decode this signal or estimate the channel. At both ends, Alice and Bob multiply the received signal by the local generated random signal to compose a shared randomness. Alice obtains $C_a = x_a x_b h_{ba}$, and Bob obtains $C_b = x_b x_a h_{ab}$. If reciprocity holds, say $h = h_{ab} = h_{ba}$, $C_a = C_b$, which can be used to generate a shared key.

Under the manipulative jamming attack, assuming the original channel probing signal is destroyed by the attacker, what Alice and Bob receive will be the attacking signals with the same power. Alice and Bob will multiply locally generated random signals by the received attacking signal. Alice and Bob can still generate keys from the measurements when $x_a = x_b$ at a 50 percent chance if x_a and x_b are binary. The generated keys, however, will not be compromised by Eve since she has no knowledge of x_a or x_b.

Under the channel manipulation attack, although the attacker can control the channel h, it will not lead to a compromised key. Under this attack, Alice and Bob are still able to generate a shared key based on $x_a x_b h$, which is different from h.

The methodology of integrating user-introduced randomness and channel reciprocity to generate keys is analogous to D-H key exchange. The independent random signals generated at both sides are analogous to the random numbers generated in D-H protocol. Users multiplying received signals by locally generated random signals to achieve agreement is analogous to users conducting discrete exponential operation on the received numbers with locally generated random numbers in D-H protocol. Note that all four steps after channel probing illustrated in Fig. 1 are still needed in order to generate a shared key for the proposed method.

Security Analysis under Passive and Active Attacks

We now compare the security strength of the traditional key generation scheme that uses constant probing signals with the proposed one that exploits random probing signals under both passive and active attacks. Two metrics are used:

- *Key generation rate* measuring how much secret mutual information is shared between Alice and Bob given the information obtained by Eve
- *Leaked information* measuring mutual information among Alice's received signals, Bob's received signals, and Eve's overheard (for passive attacks) or injected signals (for active attacks)

For passive attacks, we assume a case when Eve is close to Alice. We set the correlation coefficient between legitimate channel and eavesdropping channel as 0.9. For disruptive jamming cases, we assume the attacker injects a constant signal. For manipulative jamming cases, we assume that the attacker always chooses the best attacking moment when the attacking channels have equal gain. For channel manipulation cases, we assume that the attacker can control the channel coefficient h.

For simplicity, we assume that the channel measurements are independent and channel reciprocity holds. We only consider the in-phase component of signals. We assume that all the channel coefficients follow zero mean Gaussian with variance of 10. All the measurement noises follow zero mean Gaussian with unit variance. Therefore, the SNR of each channel is 10 dB by default. For passive attacks, we examine the performance under different SNRs of the legitimate channel h. For active attacks, we fix the SNR of the legitimate channel as 10 dB and change the power (SNR) of the attacking signals.

The key generation rate and leaked information for different cases are shown in Figs. 3 and 4. Each point on the curve represents the numerical result with 10^6 samples. P represents a passive attack, DJ represents a disruptive jamming attack, MJ represents a manipulative jamming attack, CM represents a channel manipulation attack, C represents a key generation scheme with constant probing frame, and R represents key generation scheme integrating user-generated randomness.

From Fig. 3, we can see that under passive attacks, the proposed scheme using random probing signals achieves a similar key generation rate to the scheme using constant probing signals.

Under disruptive jamming attack, the proposed scheme achieves a much higher key generation rate than does the constant probing scheme. An interesting observation is that under disruptive jamming attack, when the SNR of the attacking signal is increased from 0 to 10 dB, the key generation rate of DJ-R is decreased, while when it keeps increasing, the key generation rate is increased. The reason behind this observation is that when the attacking signal is weak, the legitimate channel gain dominates the channel measurement. Thus, the attacking signal is considered noise, which will degrade the key generation rate. When the attacking signal becomes stronger, it dominates the channel measurement and makes channel measurements at Alice and Bob more similar, which leads to a higher key generation rate.

It can be seen that the key generation rate is nearly zero under all the active jamming attacks when constant probing signals are used. However, under active attacks, the proposed scheme achieves a much higher key generation rate. The key generation rate of CM-R is comparable to that achieved under P-R and P-C.

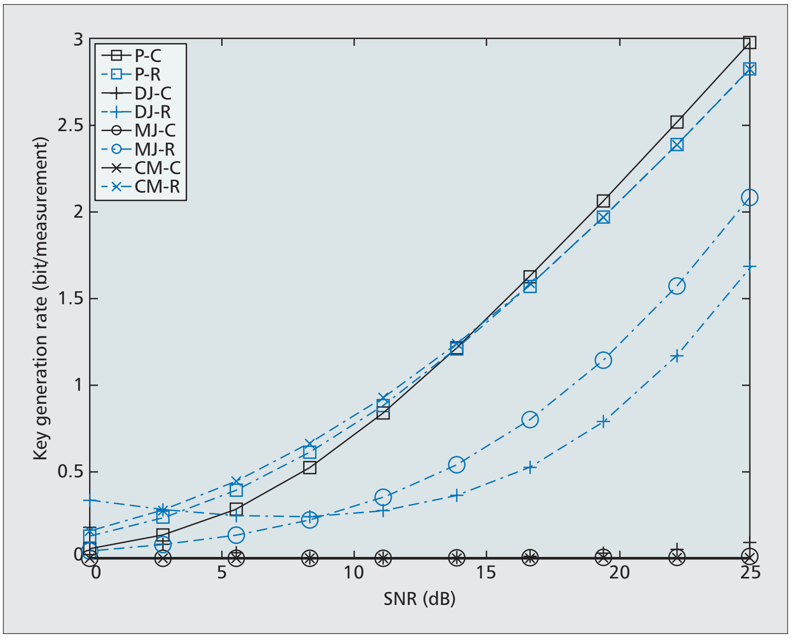

Figure 3. Key generation rate under different attack models.

The fundamental reason is that the proposed scheme integrates user-generated randomness and channel randomness to generate the key. The attacker has no control over the user-generated randomness, and thus cannot gain much information about the generated key. When the attacking signal becomes stronger, it actually helps to increase the key generation rate, since it introduces more randomness and higher reciprocity into the channel.

From Fig. 4, we can see that a large amount of information is leaked to an active attacker under manipulative jamming and channel manipulation attacks when using constant probing signals, which make the key generation rate nearly zero. When the attacking signal is stronger, more information is leaked. The leaked information here is actually the compromised key information. If Alice and Bob are not aware of these two kinds of attacks, they would generate a key that is largely controlled by the attacker.

When using random probing signals, the amounts of leaked information to both passive and active attackers under different attacking models are small, which indicates a strong security strength of the proposed scheme. It can be noted that although the scheme is proposed to counter active attacks, it also improves the security strength under passive attacks. As shown in Fig. 4, under passive attacks, the leaked information using random probing signals is much less than that using constant probing signals.

Future Research Topics

We have discussed various practical challenges of physical layer key generation and analyzed its security strength under both active and passive

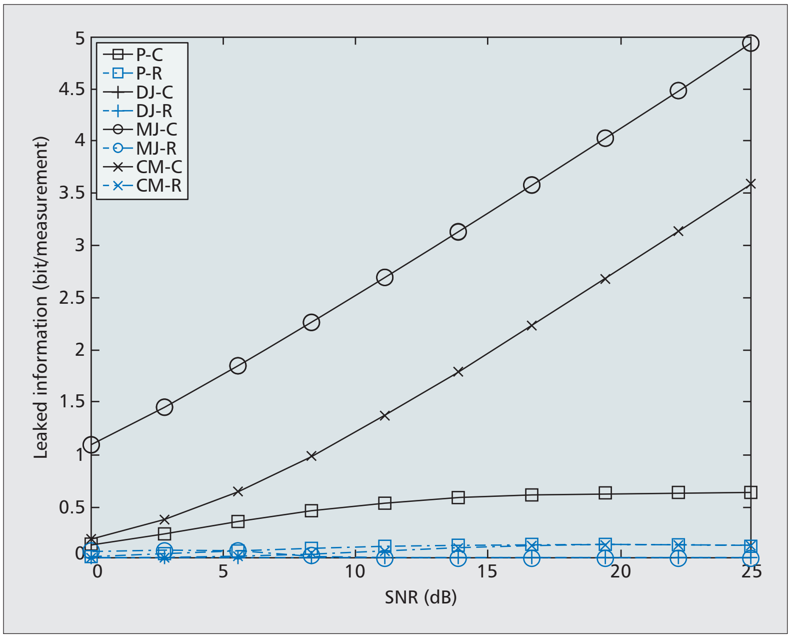

Figure 4. Leaked information under different attack models.

attacks. A lot of research issues and opportunities remain for further investigation. We discuss some of them in the following subsections.

Multi-User Case

Most of the existing key generation schemes are designed for two-user scenarios. Although the multi-user case is common in practice, the research on multi-user key generation is largely open. Both theoretical analysis on the secret key capacity and the design of practical protocols need further investigation. Users may need to generate pair-wise keys or group keys [12] under various constraints of time delay, energy consumption, spectrum efficiency, and so on. How to schedule the channel probing in a multi-user scenario is an interesting issue. In general, when a user probes the channel faster, it can achieve higher key generation rate [4]. However, the entropy of each measurement is reduced due to increased correlation between the consecutive channel measurements. Each user may alternatively probe the channel in order to fairly share the medium. The broadcast nature of the wireless channel can also be exploited to reduce the probing overhead if one user needs to generate a shared key with multiple neighbors.

Multihop Networks

Multihop networks such as mesh networks, mobile ad hoc networks, and sensor networks require secure end-to-end communication via multiple hops. An intuitive extension of the single-hop physical layer key generation to multihop networks is to generate a pair-wise secret key on each link and generate an end-to-end shared secret through per-hop encryption. However, an intermediate node or a forwarder may not be trustworthy or be compromised. There may be multiple paths between two nodes. The security strength, energy consumption, and delay on each path may be different. Generating an end-to-end secret key at the physical layer with consideration of joint medium access control and routing design under untrustworthy intermediate nodes is worth further investigation. Multipath diversity can also be exploited to enhance the security strength of the generated key.

Full-Duplex Radio

Almost all the existing physical layer key generation research is based on the assumption of half-duplex radio. With the practical implementation of full-duplex radio becoming reality, physical layer key generation can be made more efficient. A straightforward benefit is the decreased time lag between two bidirectional channel probings, which will increase the reciprocity of the channel measurements. When Alice and Bob transmit the probing signal at the same time, a passive attacker may only overhear a superposed signal, which prevents the attacker gaining CSI, thus enhancing security. However, an advanced attacker with multiple antennas may perform beamforming to separate the probing signals to gain CSI. Furthermore, the attacker can also jam and listen at the same time with full-duplex capability, which may increase the chance of key compromise for manipulative jamming attacks. The security strength of the proposed key generation scheme with random probing signals is expected to be maintained as long as the attacker cannot figure out the random probing signals. Near-field communication that exploits inductive coupling naturally supports full-duplex communication, which can also be exploited to achieve low-cost high-throughput key generation [13].

Millimeter-Wave Communication and Massive MIMO

With the evolution to 5G cellular communications, future wireless devices will be equipped with 60 GHz millimeter-wave radios with tens of antennas. The propagation property of millimeter-wave is different from those at lower frequency, and is more like a beam. This unique propagation property may be exploited to enhance physical layer key generation security since the channel may get decorrelated within a very short distance, thus improving security strength under eavesdropping attacks. With many antennas, the key generation rate is expected to be significantly increased with the increase of antenna spatial diversity. On the other hand, the channel coherence time, which is usually inversely proportional to the carrier frequency, will be significantly small, which introduces challenges on ensuring reciprocity of bidirectional channel measurements. Full-duplex technology may be integrated with millimeter-wave communication to achieve high channel reciprocity.

Integrating other Physical/Physiological Random Sources

Other than using physical layer information to generate shared keys, we can integrate other random sources, such as readings from co-located sensors [14] or physiological information [15]. The IoT will consist of various devices equipped with sensors and wireless interfaces. How to exploit the properties of sensor readings and channel randomness to bootstrap device-to-device secure communication and at the same time protect user privacy is an interesting issue.

Conclusion

In this article, we provide an overview of the physical layer key generation process and point out its practical challenges and possible solutions. We analyze the security strength of physical layer key generation under different attacking models. In order to achieve secure physical layer key generation under active attacks, we propose the use of random probing signals to hide the channel state information, and combine user generated randomness and channel randomness to generate a shared secret key. Numerical results show that the proposed scheme achieves much higher security strength than the existing scheme using constant probing signals. Future research directions on physical layer key generation under multi-user multihop scenarios and new communication technologies including full-duplex radio and millimeter-wave communications are discussed. It is also promising to integrate other physical or physiological information to generate shared secret keys.

Acknowledgments

This research was partially supported by the NSF CAREER award under Grant Number (CNS-1502584) and NSF EARS program grant under Grant Number (CNS-1464487).

References

[1] J. Hershey, A. Hassan, and R. Yarlagadda, "Unconventional Cryptographic Keying Variable Management," *IEEE Trans. Commun.*, vol. 43, no. 1, Jan 1995, pp. 3–6.

[2] S. Jana *et al.*, "On the Effectiveness of Secret Key Extraction from Wireless Signal Strength in Real Environments," *Proc. 15th ACM Annual Int'l. Conf. Mobile Computing and Networking '09*, 2009, pp. 321–32.

[3] N. Patwari *et al.*, "High-Rate Uncorrelated Bit Extraction for Shared Secret Key Generation from Channel Measurements," *IEEE Trans. Mobile Computing*, vol. 9, no. 1, Jan. 2010, pp. 17–30.

[4] Y. Wei, K. Zeng, and P. Mohapatra, "Adaptive Wireless Channel Probing for Shared Key Generation Based on Pid Controller," *IEEE Trans. Mobile Computing*, vol. 12, no. 9, Sept. 2013, pp. 1842–52.

[5] S. Ali, V. Sivaraman, and D. Ostry, "Eliminating Reconciliation Cost in Secret Key Generation for Body-Worn Health Monitoring Devices," *IEEE Trans. Mobile Comp.*, vol. 13, no. 12, Dec. 2014, pp. 2763–76.

[6] K. Zeng *et al.*, "Exploiting Multiple-Antenna Diversity for Shared Secret Key Generation in Wireless Networks," *Proc. IEEE INFOCOM '10*, 2010, pp. 1837–45.

[7] U. M. Maurer, "Secret Key Agreement by Public Discussion from Common Information," *IEEE Trans. Info. Theory*, vol. 39, no. 3, May 1993, pp. 733–42.

[8] M. Edman, A. Kiayias, and B. Yener, "On Passive Inference Attacks against Physical-Layer Key Extraction," *Proc. 4th Euro. Wksp. System Security*, 2011, pp. 8:1–8:6.

[9] K. Argyraki *et al.*, "Creating Secrets Out of Erasures," *Proc. 19th Annual Int'l. Conf. Mobile Computing and Networking*, 2013, pp. 429–40.

[10] M. Zafer, D. Agrawal, and M. Srivatsa, "Limitations of Generating a Secret Key Using Wireless Fading under Active Adversary," *IEEE/ACM Trans. Networking*, vol. 20, no. 5, Oct. 2012, pp. 1440–51.

[11] S. Eberz *et al.*, "A Practical Man-In-The-Middle Attack on Signal-Based Key Generation Protocols," *Proc. ESORICS, LNCS*, S. Foresti, M. Yung, and F. Martinelli, Eds., vol. 7459. Springer, 2012, pp. 235–52.

[12] H. Liu *et al.*, "Group Secret Key Generation via Received Signal Strength: Protocols, Achievable Rates, and Implementation," *IEEE Trans. Mobile Computing*, vol. 13, no. 12, Dec. 2014, pp. 2820–35.

[13] R. Jin et al., "Practical Secret Key Agreement for Full-Duplex Near Field Communications," *Proc. 9th ACM Symp. Info., Comp. and Commun. Security*, 2014, pp. 217–28.

[14] V. M. Prabhakaran, K. Eswaran, and K. Ramchandran, "Secrecy via Sources and Channels," *IEEE Trans. Info. Theory*, vol. 58, no. 11, Nov. 2012, pp. 6747–65.

[15] K. K. Venkatasubramanian *et al.*, "Pska: Usable and Secure Key Agreement Scheme for Body Area Networks," *IEEE Trans. Info. Tech. Biomed.*, vol. 14, no. 1, Jan. 2010, pp. 60–68.

Biography

Kai Zeng is an assistant professor in the Department of Electrical and Computer Engineering at George Mason University. He received his Ph.D. degree in electrical and computer engineering from Worcester Polytechnic Institute (WPI) in 2008. He was a postdoctoral scholar in the Department of Computer Science at University of California, Davis (UCD) from 2008 to 2011. He worked in the Department of Computer and Information Science at the University of Michigan — Dearborn as an assistant professor from 2011 to 2014. He was a recipient of the U.S. National Science Foundation Faculty Early Career Development (CAREER) award in 2012. He won the Excellence in Postdoctoral Research Award at UCD in 2011 and Sigma Xi Outstanding Ph.D. Dissertation Award at WPI in 2008. He is an Editor of *IEEE Transactions on Wireless Communications*. His current research interests are in cyber-physical system security and privacy, physical layer security, network forensics, and cognitive radio networks.

The IoT will consist of various devices equipped with sensors and wireless interfaces. How to exploit the properties of sensor readings and channel randomness to bootstrap device-to-device secure communication and at the same time protect user privacy is an interesting issue.