ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 4, pp: (149-153), Month: October - December 2015, Available at: www.researchpublish.com

Physical Layer Wireless Security Shape with Noisy Symbols

[1]V.Prabhu, [2]P.Murugan, [3]P. Kuppusamy

[1]Gnanamani College of Technology
[2]Assistant Professor, Gnanamani College of Technology
[3]Head of Department, Gnanamani College of Technology

Abstract: **In current scenario we present our approach regarding the implementation of new wireless security shaped with noisy symbols, using a software-development driven approach. Communication security is critical and increasingly challenging issue in wireless networks. We propose a multiple inter-symbol obfuscation (MIO) scheme, which utilizes a set of artificial noisy symbols (symbols key) to obfuscate the original data symbols in the physical layer. MIO can effectively enhance the wireless communications security. On the one hand, an eavesdropper, without knowing the artificial noisy symbols, cannot correctly decrypt the obfuscated symbols from the Eavesdropped packets. On the other hand, a legitimate receiver can easily check the integrity of the symbols key and then reject the fake packets from the received packets. The security analysis reveals that, without considering the initial key, the MIO scheme can achieve information-theoretic secrecy against the passive eavesdropping attack and computational secrecy against the fake packet injection attack.**

Keywords: **artificial noisy symbols (symbols key), Multiple inter-symbol obfuscation (MIO).**

1. INTRODUCTION

The past few years have produced innovative health-oriented networking and wireless communication technologies, ranging from Low-power medical radios that harvest body energy to wireless sensor networks for in-home monitoring and diagnosis. Today, such wireless systems have become an intrinsic part of many modern medical devices. In particular, implantable medical devices (IMDs), including pacemakers, cardiac defibrillators, insulin pumps, and neurostimulators all feature wireless communication. Adding wireless connectivity to IMDs has enabled remote monitoring of patients' vital signs and improved care providers' ability to deliver timely treatment, leading to a better health care system.

In the past, real-world prototyping of communication systems was not very common due to high financial costs and long development times derived from the use of dedicated hardware (HW) such as field programmable gate arrays (FPGA), application specific integrated circuits (ASICs) and digital signal processors (DSPs). Nonetheless, this implementation is vital because it allows the exposure of problems, such as unexpected system behavior, real-world impairments and design flaws. In spite of this, a HW based approach also suffers From lack of flexibility and modularity, which on the other hand are highly desirable qualities. These qualities are provided by SDR platforms, where physical layer (PHY) and medium access control MAC) functions are performed in software in a General Purpose Processor (GPP), while only the radio frequency (RF) and signal conversion functions such as sampling and own conversion are performed in programmable hardware. Moreover, these platforms use, as development tools, popular high level programming languages such as C/C++, which present the advantage of being dominant Programming languages, so a great number of experienced programmers and a number of well-written peer-reviewed libraries are available, which open the door to create communities of software radio developers.

Inalterability: In the U.S. alone, there are millions of people who already have wireless IMDs and about 300,000 such IMDs are implanted every year. Once implanted, an IMD can last up to 10 years, and replacing it requires surgery that carries risks of major complications. Incorporating cryptographic mechanism into existing IMDs may be infeasible because of limited device memory and hence can only be achieved by replacing the IMDs.

This is not an option for people who have IMDs or may acquire them in the near future.

Safety: It is crucial to ensure that health care professionals always have immediate access to an implanted device. However, if cryptographic methods are embedded in the IMD itself, the device may deny a health care provider access unless she has the right credentials. Yet, credentials might not be available in scenarios where the patient is at a different hospital, the patient is unconscious, or the cryptographic key storage is damaged or unreachable. Inability to temporarily adjust or disable an IMD could prove fatal in emergency situations.

Maintainability: Software bugs are particularly problematic for IMDs because they can lead to device recalls. In the last eight years, about 1.5 million software-based medical devices were recalled. Between 1999 and 2005, the number of recalls of software based medical devices more than doubled; more than11% of all medical-device recalls during this time period were attributed to software failures. Such recalls are costly and could require surgery if the model is already implanted. Thus, it is desirable to limit IMDs' software to only medically necessary functions.

II. THREAT MODEL

The wireless communications security is to prevent attackers from intercepting the wireless communications, while still delivering contents to the intended recipients. In this paper, we address two types of adversaries, passive Eavesdropping attack and fake packet injection attack, during the wireless communications, just like some former works:

1) Passive Eavesdropping Attack: An adversary eavesdrops on the wireless medium and intercepts the wireless transmission between the legitimate transmitter and receiver. It can attempt to decode the signal from the intercepted signal with the presence of the MIO scheme. The MIO scheme will provide the information-theoretic secrecy to enhance the wireless communications security.

2) Fake Packet Injection Attack: An adversary injects fake packets to the legitimate users, triggering the events to further.

Disrupt the user's manner (e.g., mislead the users' operations).Unlike the passive eavesdropping attack, it can deploy the brute-force to test all possible symbols keys to inject a fake packet. The MIO scheme will enhance the computational secrecy to defend against this attack. However, we do not consider the cases where the legitimate transmitter or receiver is physically compromised because the data confidentiality is no longer ensured no matter what security measure is adopted to secure the wireless communications between two hosts if any one of them is not secured. Additionally, we do not consider the jamming-based denial of service (DoS) attack in this paper, where the adversary simply jams the channel with extraordinary transmission power, since the legitimate sender and receiver fail to communicate with each other under this DoS attack.

III. SYSTEM DESIGN

This section provides the design of the multiple Inter-symbol obfuscation (MIO) which includes two stages:

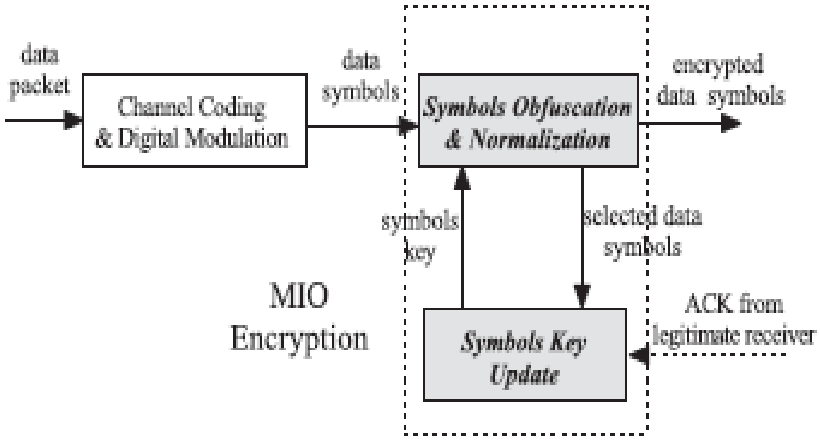
MIO encryption (adding the artificial noisy symbols key), and MIO decryption (offsetting the artificial noisy symbols key).

Although the MIO scheme is designed based on the multiple inter-symbol obfuscation at the physical layer, it still needs an initial key to start the secure wireless communications. For ease of presentation, Table I lists the notations used in the following sections.

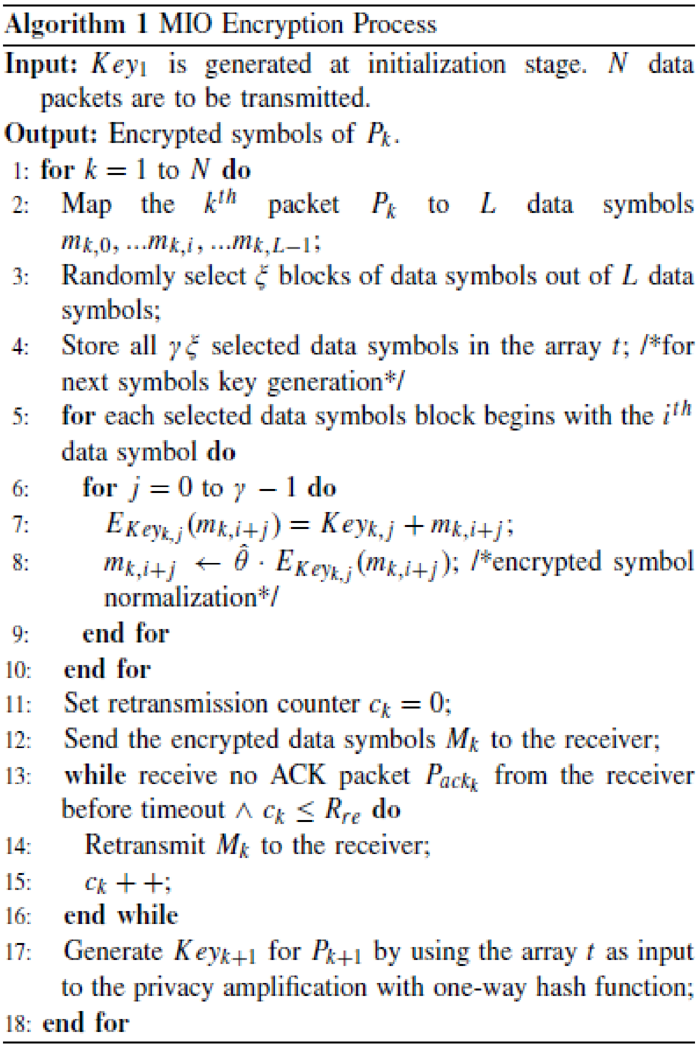
A. Initialization:

To initiate the first symbols key in a non-secure wireless channel, we first take the conventional key agreement protocols, e.g., EKE or augmented EKE, to achieve a bit-level authenticated key. Then, the bit-level authenticated key can be used to generate parameters by a one-way harsh function. After that, the parameters, which include the size of the symbols key γ, the angle between the key symbol and the Real-axis Vk, j and the magnitude ratio of the key symbol and unit-power symbol α, are used to generate the first symbols key without any trusted third party. Obviously, the legitimate transmitter and receiver have to exchange some redundant packets and deploy the same set of hash functions to generate these parameters.

As the bit-level key agreement schemes can only provide computational secrecy but not information-theoretic secrecy, the key can be compromised if the eavesdropper has enough computational power (detailed in Section V-A). Moreover, the initial key still requires the legitimate transmitter and receiver



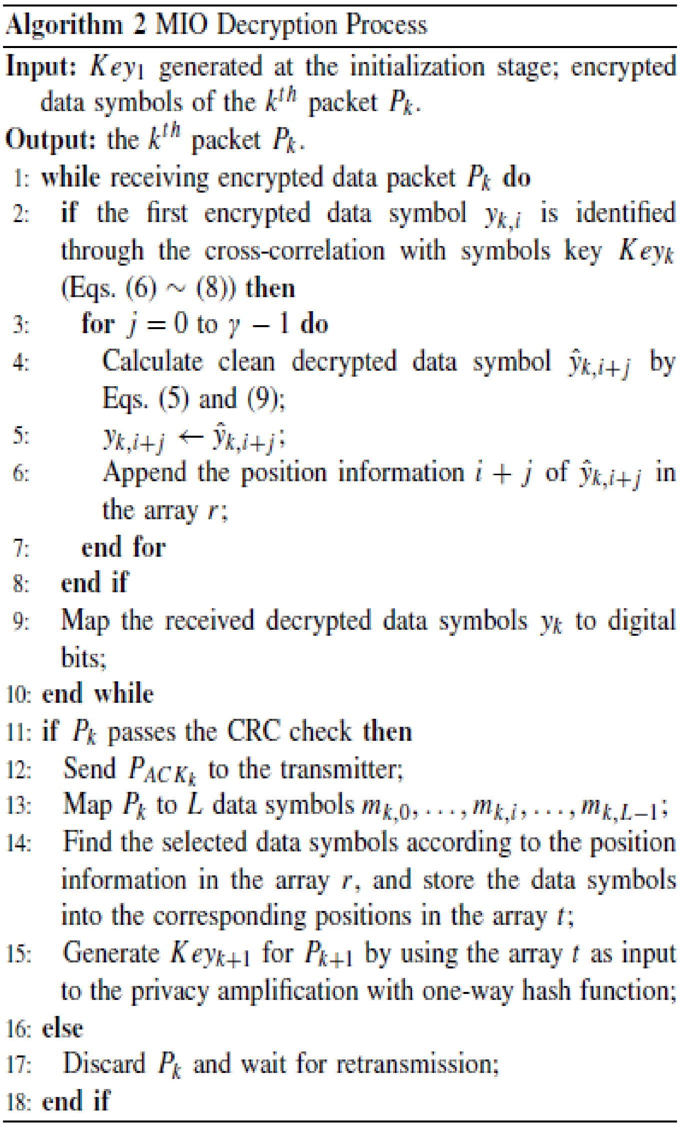
The MIO encryption process at the legitimate transmitter.



```
Algorithm 1 MIO Encryption Process
Input: $Key_1$ is generated at initialization stage. $N$ data
    packets are to be transmitted.
Output: Encrypted symbols of $P_k$.
for $k = 1$ to $N$ do
    Map the $k^{th}$ packet $P_k$ to $L$ data symbols
    $m_{k,0}, ...m_{k,i}, ...m_{k,L-1}$;
    Randomly select $\xi$ blocks of data symbols out of $L$ data
    symbols;
    Store all $\gamma\xi$ selected data symbols in the array $t$; /*for
    next symbols key generation*/
    for each selected data symbols block begins with the $i^{th}$
    data symbol do
        for $j = 0$ to $\gamma - 1$ do
            $E_{Key_{k,j}}(m_{k,i+j}) = Key_{k,j} + m_{k,i+j}$;
            $m_{k,i+j} \leftarrow \hat{\theta} \cdot E_{Key_{k,j}}(m_{k,i+j})$; /*encrypted symbol
            normalization*/
        end for
    end for
    Set retransmission counter $c_k = 0$;
    Send the encrypted data symbols $M_k$ to the receiver;
    while receive no ACK packet $P_{ack_k}$ from the receiver
    before timeout $\wedge$ $c_k \leq R_{re}$ do
        Retransmit $M_k$ to the receiver;
        $c_k + +$;
    end while
    Generate $Key_{k+1}$ for $P_{k+1}$ by using the array $t$ as input
    to the privacy amplification with one-way hash function;
end for
```

We first consider that legitimate transmitter A is about to send N data packets to legitimate receiver B. As shown in Fig, for each data packet, it goes through the MIO encryption process by two steps

1. symbols obfuscation
2. Normalization and symbols key update at the transmitter.



Algorithm 2 MIO Decryption Process

```
Input: $Key_1$ generated at the initialization stage; encrypted data symbols of the $k^{th}$ packet $P_k$.
Output: the $k^{th}$ packet $P_k$.
while receiving encrypted data packet $P_k$ do
    if the first encrypted data symbol $y_{k,i}$ is identified through the cross-correlation with symbols key $Key_k$ (Eqs. (6) $\sim$ (8)) then
        for $j = 0$ to $\gamma - 1$ do
            Calculate clean decrypted data symbol $\hat{y}_{k,i+j}$ by Eqs. (5) and (9);
            $y_{k,i+j} \leftarrow \hat{y}_{k,i+j}$;
            Append the position information $i + j$ of $\hat{y}_{k,i+j}$ in the array $r$;
        end for
    end if
    Map the received decrypted data symbols $y_k$ to digital bits;
end while
if $P_k$ passes the CRC check then
    Send $P_{ACK_k}$ to the transmitter;
    Map $P_k$ to $L$ data symbols $m_{k,0}, \ldots, m_{k,i}, \ldots, m_{k,L-1}$;
    Find the selected data symbols according to the position information in the array $r$, and store the data symbols into the corresponding positions in the array $t$;
    Generate $Key_{k+1}$ for $P_{k+1}$ by using the array $t$ as input to the privacy amplification with one-way hash function;
else
    Discard $P_k$ and wait for retransmission;
end if
```

It is noted that in the MIO decryption process, after filtering noises and channel coefficients, the digital bits which are mapped into the data symbols for the key updating are exactly the same as those for the transmitter. Associating the selected data symbols with their position information in the array r, the receiver can store the corresponding selected data symbols in the array t. Thus, it would guarantee that the array t at the receiver for the key updating is the same as the array at the transmitter. The MIO decryption process algorithm is shown in Algorithm 2.

IV. CONCLUSIONS

In this paper, we propose a multiple inter-symbol obfuscation (MIO) scheme to secure the wireless transmissions between two legitimate entities. MIO does not need any trusted third party to interfere the packet interception by the eavesdropper or static channel condition to cancel artificial noises. Rather, it employs the data symbols from the previous data packets to generate the symbols key which obfuscates the current data packets. By dynamically updating the symbols key as the packets are disseminated, it is hard for an adversary to brute-force the symbols key by intercepting a number of encrypted symbols and analyzing them off-line. We establish the mathematical model for MIO, and prove that MIO can provide both the information-theoretic secrecy and computational secrecy without considering the initial key. Additionally, the experimental results reveal that without knowing the symbols key, the BER in the MIO scheme can effectively ruin the packet reception at the eavesdropper side, and the key checking process would defend against the packet injection attack in wireless networks.

REFERENCES

[1] R. Farrell, M. Sanchez, and G. Corley, “Software-Defined Radio Demonstrators: An Example and Future Trends”, International Journal of Digital Multimedia Broadcasting, vol. 2009, Article ID 547650, 12 pages, 2009,doi:10.1155/2009/547650.

[2] Universal Software Radio Platform, Ettus Research LLC, http://www.ettus.com

[3] “GNU Radio, the GNU Software Radio project, http://gnuradio.org/redmine/wiki/gnuradio

[4] Physical Layer for Dynamic Spectrum Access and Cognitive Radio, http://www.ict-phydyas.org/

[5] 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 Proc. 15th Annu. Int.Conf. ACM MobiCom, Sep. 2009, pp. 321–332.

[6] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, “They can hear your heartbeats: Non-invasive security for implantable medical devices,” in Proc. ACM SIGCOMM, Aug. 2011, pp. 2–13.

[7] Y.-S. Shiu, S. Y. Chang, H.-C. Wu, S. C.-H. Huang and H.-H. Chen, “Physical layer security in wireless networks: A tutorial,” IEEE Wireless Commun., vol. 18, no. 2, pp. 66–74, Apr. 2011.