Journal of Advanced Research in Embedded System
Volume 7, Issue 3&4 - 2020, Pg. No. 24-28
Peer Reviewed Journal

Article

A Light-Weight White-Box Encryption Scheme for Securing Distributed Embedded Devices

Shridevi[1], Veerendra Dakulagi[2]

[1]Student, [2]Associate Professor and Dean, Department of Electronics and Communication Engineering, Guru Nanak dev Engineering College Bidar, Karnataka.

INFO

Corresponding Author:
Shridevi, Department of Electronics and Communication Engineering, Guru Nanak dev Engineering College Bidar, Karnataka.
E-mail Id:
Sridevisj180@gmail.com

How to cite this article:
Shridevi, Dakulagi V. A Light-Weight White-Box Encryption Scheme for Securing Distributed Embedded Devices. *J Adv Res Embed Sys* 2020; 7(3&4): 24-28.

Date of Submission: 2020-08-29
Date of Acceptance: 2020-09-13

ABSTRACT

Embedded devices distributed widely in sensor networks and the Internet of Things is used to collect and send data. Many of them are deployed unattended (for example, sensor nodes and tag readers), while others can be easily lost (for example, bracelets and smart watches). This distributed hardware can be captured and accessed unauthorized due to its physical nature. From a security perspective, they generally work in the context of a white square attack, where opponents have full visibility of deployed cryptographic systems deployments and full control over their implementations. Undoubtedly, it is a big challenge to deal with white box attacks on these devices. Current encryption algorithms for white box attack contexts require a large amount of memory space and are therefore unsuitable for embedded devices with limited resources. To meet this challenge, we suggest a new lightweight encryption plan to protect data confidentiality. Encryption is done using specialized confidential components and the encryption algorithm requires a small amount of static data to store critical information. Also, this scheme supports only effective key update at a very small COST. The safety and cost of the proposed scheme have been theoretically analyzed with positive results and extensive empirical assessments indicate that the new scheme meets the requirements of distributed hardware combined in terms of limited memory usage and low arithmetic cost.

Keywords: White-Box Attack Context, Distributed Embedded Devices, General Public License

Introduction

IoT systems, embedded devices, such as sensor nodes, tag readers, smart wristbands and watches, are commonly executed for gathering and sending information. Since the acquired data, which are typically sensitive, will be transferred to remote servers, such embedded devices necessitate protection measures for data confidentiality.[1-3] Without effective protection, business secrets could be exposed, and personal privacy could be invaded. The two major techniques for data encryption are secret-key (symmetric) encryption and public-key (asymmetric) encryption, where the former is more suitable for distributed embedded devices, because the latter usually requires costly mathematical computations such as manipulations of points on elliptic curves, which may not fit resource-constrained embedded devices. Unfortunately, the standard design and implementation of secret-key encryption algorithms do not take into consideration the scenario where an adversary can capture the device and have total visibility of the implementation of the built-in cryptosystem and full control over its execution. Such scenario is called the White-Box Attack Context (WBAC), or more abstractly, we say that the cryptosystem runs in the white-box attack model.[4-6] In this scenario, the adversary can extract cryptographic primitives and obtain unlimited access to the information stored in the device's memory chip. Such attack can be conducted by applying reverse engineering and probing techniques to chip-level components of the device.[7-9] Consequently, the secret cryptographic key can be extracted from the implementation, and thus the encrypted data acquired by eavesdropping the communication can be recovered. This is a critical security threat to IoT systems, particularly for sensitive data congregated and distributed by distributed embedded devices.[3,10] Although some white-box encryption algorithms (WBEAs) have been developed to deliver practical protection for the applications of out-dated symmetric encryption algorithms such as AES and DES, most of the known WBEAs have been threatened by both specialized attacks[11-15] and general attacks.[16-18] To confront this dilemma, in this paper, we suggest a novel symmetric encryption scheme, where the encryption can be achieved securely in WBACs, with the resulting salient features. First, it only involves a limited number of straightforward procedures such as table look-ups and bit operations because the processor of a distributed embedded device is usually not powerful. Second, in our scheme, the size of static data, which contains critical information (e.g., lookup tables) for encryption process, is small so that it can run in resource-constrained embedded devices. Third, it achieves medium-level security with a reasonable protection duration. That is, adversaries could not break the scheme in short time. Fourth, it supports efficient key-updating at ridiculously small cost. We devise this new scheme by adapting the secret structures based on the Feistel network with a pair of extra inserted bijections. Moreover, our new scheme integrates computational components of different width together to defeat structural cryptanalysis. In summary, our major contributions are as follows: (a) We propose a novel white-box encryption scheme with reasonable- security. A salient feature of our scheme lies in the support of efficient key-updating at ridiculously small cost, which has been overlooked in most related studies. (b) The new white-box encryption scheme has fast encryption speed, and the size of static data is relatively small, making it very suitable for resource-constrained distributed embedded devices.

Literature Review

Sweet Dreams and Nightmares: Security in the Internet of Things.

Authors: *Timo Kasper, David Oswald, Christ of Paar*

Wireless embedded devices are key on the Internet of Things: Objects marked with Radio Frequency IDentification and Near Field Communication skill, smartphones, and other embedded tokens interact from device to device and thereby often process information that is security or privacy relevant for humans. For protecting sensitive data and preventing attacks, many embedded devices employ cryptographic algorithms and authentication schemes. In the past years, various vulnerabilities have been found in commercial products that enable to bypass the security mechanisms. Since many the devices in the field are in the hands of potential adversaries, application attacks (such as side-channel analysis and reverse engineering) can play a crucial role for the overall security of a system. At hand of numerous examples of assailable industrial goods, we explain the potential influence of the found security weaknesses and illustrate "how to not do it".

"Social Access vs. Privacy in Wearable Computing: A Case Study of Autism,"

Authors; *R. Kirkham, C. Greenhalgh*

People with high-performance autism face questions in communication and social interaction. This article thinks the prospect, and perhaps certainty, of apparel devices such as Google Glass being used as real-time assistive experiences for this group, with the intent of enabling them to better access our complex social world. Social impairments, by their very nature, feature issues of interaction, personal information, and social judgment. In pondering such assistive expertise in this context, the authors explore new tensions between privacy issues and assistive tools, particularly those of a do-it-by hand nature, which are not instantly solvable within our current privacy frameworks. This editorial is part of a particular issue on privacy and security.

Design Approach

In this chapter we discuss about the block diagram of A Light-Weight White-Box Encryption Scheme for Securing Distributed Embedded Devices hardware requirements and software requirements.

Block Diagram of the Implementation of a Light

Weight White-Box Encryption Scheme for Securing Distributed Embedded Devices

Contains transmitter and receiver sections the below figure shows the transmitter block diagram of A Light-Weight White-Box Encryption Scheme for Securing Distributed Embedded Devices Figure 1.

Working of the Project:

In this project our main aim is to apply encryption algorithm to embedded application. Here we will take four sensor data and need to transmit to receiver section while sending we will apply encryption algorithm to encrypt the sensor data. Similarly, we need to receive the sensor data using receiver and we need to apply decryption algorithm to decrypt sensor data. Here we are using zigbee for wireless data transmission.

Transmitter

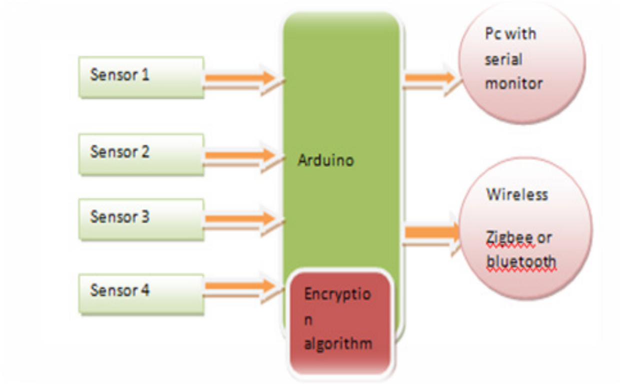


Figure 1.Block Diagram

Receiver

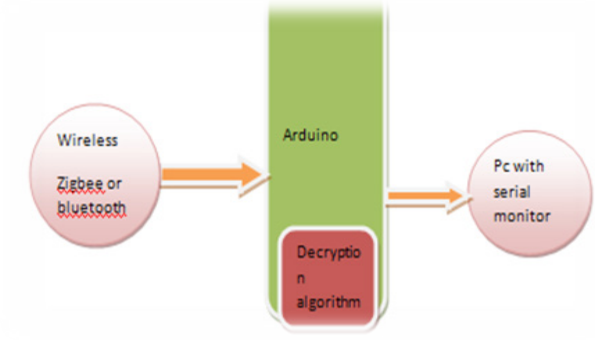


Figure 2.Arduino Pin Diagram

Arduino Uno Micro Controller

Arduino uno microcontroller is an open source, computer hardware and software company, development, and user group that designs and manufactures single-board microcontrollers and microcontroller kits for constructing digital devices and interactive objects that can sense and control objects in the natural world. The paper's products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL),[1] facilitating the production of Arduino boards and software delivery by anyone. Arduino boards are accessible commercially in prefabricated form, or as do-it-by hand kits Figure 2.

The input to the circuit is employed from the regulated power supply. The A.C input i.e., 230V from the mains supply is step down by the transformer to 12V and is fed to a rectifier. The output obtained from the rectifier is a thumping D.C voltage. So, to get a pure D.C voltage, the output voltage from the rectifier is fed to a filter to get rid of any A.C elements present even after rectification. Now, this voltage is given to a voltage regulator to attain a pure constant dc voltage Figure 3 and Figure 4.

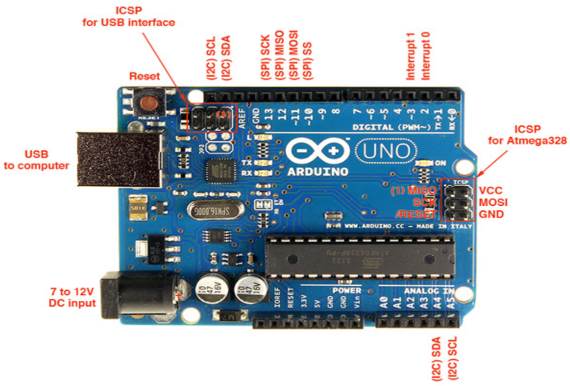


Figure 3.Power Supply Unit

Usually, D.C voltages are required to manage various electronic equipment and these voltages are 5V, 9V or 12V. But these voltages cannot be discovered immediately.

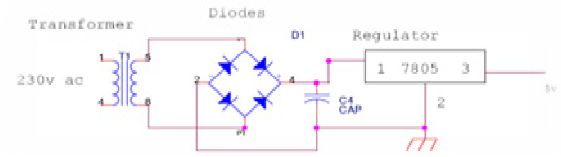


Figure 4.Step-down Transformer

Thus, the A.C. input accessible at the mains supply i.e., 230V is to be taken down to the necessary voltage level. This is done by a transformer. Thus, a step-down transformer is working to reduce the voltage to a necessary level.

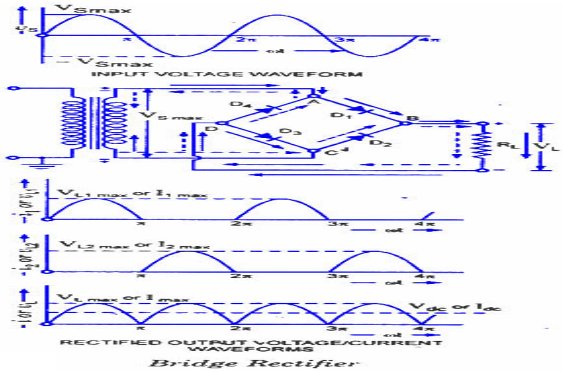


Figure 5.Output Voltage/ Current Waveforms of Bridge Rectifier

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore, a regulator is applied at the output stage. (Figure 5)

Voltage Regulator

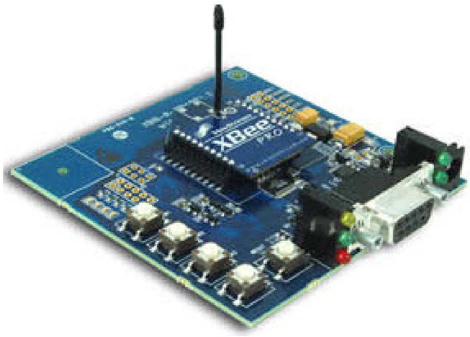
As the name itself implies, it regulates the input applied to it. A voltage regulator is an electrical regulator designed to inevitably sustain a constant voltage level. In this project, power supply of 5V and 12V are required. To obtain these voltage levels, 7805 and 7812 voltage regulators are to be used. The first number 78 represents positive supply and the numbers 05, 12 represent the required output voltage levels. The L78xx series of three-terminal positive regulators is accessible in TO-220, TO-220FP, TO-3, D2PAK and DPAK packages and some fixed output voltages, making it useful in a wide range of products. These regulators can offer local on-card regulation, eliminating the distribution difficulties linked with single point regulation. Each type uses internal current limiting, thermal shut-down and safe area safety, making it basically unbreakable.

Zigbee

In this current interaction world, there are several high data rate communication standards that are accessible, but none of these meet the sensors' and control devices' communication standards. These high-data rate communication requirements require low-latency and low-energy consumption even at lower bandwidths. The accessible commercial wireless systems' Zigbee technology is low-cost and low-power consumption and its exceptional and superb qualities makes this communication best suited for numerous embedded products, industrial control, and home automation, and so on.

What is Zigbee Technology?

Zigbee communication is expressly constructed for control and sensor networks on IEEE 802.15.4 standard for wireless personal area networks (WPANs), and it is the product from Zigbee alliance. This communication standard outlines physical and Media Access Control (MAC) layers to handle many devices at low-data rates. These Zigbee's WPANs operate at 868 MHz, 902-928MHz and 2.4 GHz frequencies. The date rate of 250 kbps is best fit for periodic as well as middle two-way transmission of data between sensors and managers.



Zigbee Modem

Zigbee is ease and low-fueled lattice network broadly sent for controlling and checking applications where it covers 10-100 meters inside the reach. This correspondence framework is more affordable and less complex than the other exclusive short-range remote sensor networks as Bluetooth and Wi-Fi.

Zigbee upholds distinctive organization arrangements for expert to dominate or dominate to slave correspondences. And furthermore, it tends to be worked in various modes accordingly the battery power is moderated. Zigbee networks are extendable with the utilization of switches and permit numerous hubs to interconnect with one another for building a more extensive zone organization.

IR Sensor

Infrared innovation tends to a wide assortment of remote applications. The principal territories are detecting and controllers. In the electromagnetic range, the infrared bit is isolated into three areas: close to infrared locale, mid infrared district, and far infrared area. The frequencies of these locales and their applications are demonstrated as follows.

Close to infrared area - 700 nm to 1400 nm - IR sensors, fiber optic.

Mid infrared area -1400 nm to 3000 nm - Heat detecting.

Far infrared area - 3000 nm to 1 mm - Thermal imaging.

The recurrence scope of infrared is higher than microwave and lesser than noticeable light.

For optical detecting and optical correspondence, photograph optics advances are utilized in the close to infrared district as the light is less mind boggling than RF when executed as a wellspring of sign. Optical remote correspondence is finished with IR information transmission for short reach applications.

Algorithm

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). It is found at least six time faster than triple DES.

A substitute for DES was required as its key size was too small. With rising computing power, it was considered vulnerable against extensive key search attack. Triple DES was intended to surmount this challenge, but it was found slow.

The characteristics of AES are as follows −

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES

- Provide full specification and design details
- Software implementable in C and Java

Operation of AES

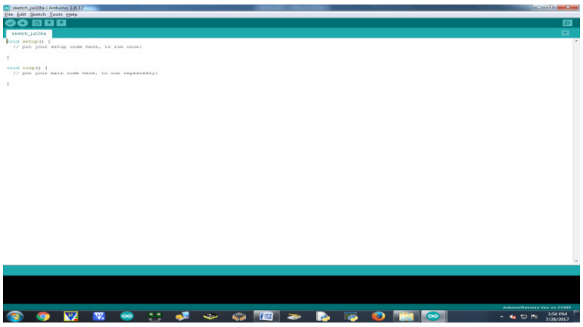
AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It encompasses of a series of linked operations, some of which involve swapping inputs by specific outputs (substitutions) and others include shuffling bits around (permutations).

Curiously, AES works all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix

Software Requirment

ARDUINO UNO R3 IDE (Integrated Development Board)

Result

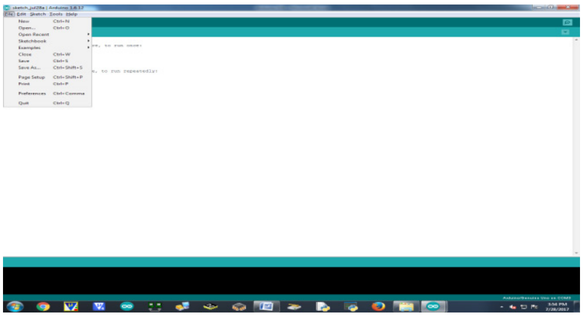


```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```



```
re, to run once:

e, to run repeatedly:
```

Conclusion

White-box attacks are critical security threats to distributed embedded devices, which could be potentially captured or accessed in an unauthorized manner. In this paper, we propose a new light-weight white-box encryption scheme, which is immune to all known attacks against WBEAs, for securing distributed embedded devices. Our new scheme is suitable for resource-constrained devices because the size of static data for encryption is small and the encryption process is efficient. Furthermore, our scheme supports efficient key-updating at very small cost, which has been usually overlooked in related studies. Extensive experimental results on real embedded devices show that our new encryption algorithm is efficient, leading to good applicability in practice.

References

1. Kasper T, Oswald D, Paar C. Sweet Dreams and Nightmares: Security in the Internet of Things. in Information Security Theory and Practice: Securing the Internet of Things, Berlin, 2014: 1-9.
2. Kirkham R, Greenhalgh C. Social Access vs. Privacy in Wearable Computing: A Case Study of Autism. *IEEE Pervasive Computing* 2015; 14(1): 26-33.
3. Zhao J. On Resilience and Connectivity of Secure Wireless Sensor Networks Under Node Capture Attacks. *IEEE Transactions on Information Forensics and Security* 2017; 12(3) : 557-571.
4. Chow S, Eisen P, Johnson H et al. A white-box DES implementation for DRM applications. *Digital Rights Management* 2003: 1-15.
5. Beunardeau M, Connolly A, Géraud R et al. White-Box Cryptography: Security in an Insecure Environment. *IEEE Security & Privacy* 2016; 14(5): 88-92.
6. Michiels W. Opportunities in White-Box Cryptography. *IEEE Security & Privacy* 2010; 1: 64-67.
7. Yum d P. J. Lee. Exact Formulae for Resilience in Random Key Predistribution Schemes. *IEEE Transactions on Wireless Communications* 2012; 11(5): 1638-1642.
8. Newell A, Yao H, Ryker A et al. Node-Capture Resilient Key Establishment in Sensor Networks: Design Space and New Protocols. *ACM Comput Sur* 2014; 47(2): 1-34.
9. Swierczynski P, Fyrbiak M, Koppe P et al. FPGA Trojans Through Detecting and Weakening of Cryptographic Primitives. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 2015; 34(8): 1236-1249.
10. Shi Y, Han J, Wang J et al. An Obfuscatable Aggregatable Signcryption Scheme for Unattended Devices in IoT Systems. *IEEE Internet of Things Journal* 2017; 4(4): 15.