The solution to improve information security for IoT networks by combining lightweight encryption protocols

Nguyen Van Tanh[1], Ngo Quang Tri[2], Mai Manh Trung[3]
[1]International School-Vietnam National University, Hanoi, Vietnam
[2]Hanoi University of Science and Technology, Hanoi, Vietnam
[3]Faculty of Information Technology- the University of Economics Technology for Industries, Hanoi, Vietnam

Article Info

Article history:

Received Feb 2, 2021
Revised Jul 15, 2021
Accepted Aug 7, 2021

Keywords:

Cyber networks security
IoT
Overhearing
Security networks
Wireless sensor networks
DLTS

ABSTRACT

With the strong development of Internet platforms today, many new technologies are applied in most areas of human social life. The development of the internet of things (IoT) has brought to the networking world many innovations with new protocols. Along with that is the complexity of security issues that greatly affect personal data, and limited resources in information technology development and applications. Many security solutions have been researched but none of them is completely effective for the IoT network because of the characteristics of energy, limited resources, processing capacity as well as operating costs. Therefore, we propose an effective solution for comprehensive protection of IoT systems against attacks to network security with improved security protocols. By combining security solutions on the layers of the IoT and code improvement, algorithmic enhancement positively contributes to this important task. In the article, we propose to improve and combine the DTLS Protocol and the overhearing mechanism and then have some experiments to prove the effectiveness, feasibility, economical and appropriate on popular IoT network models.

Corresponding Author:

Nguyen Van Tanh
International School-Vietnam National University
Hanoi, Vietnam
Email: tanhnv@vnu.edu.vn

1. INTRODUCTION

The development of the internet of things has brought to the networking world many innovations with new protocols. Along with that is the complexity of security issues that greatly affect personal data, and limited resources in information technology development and applications. Many security solutions have been researched but none of them is completely effective for the IoT network because of the characteristics of energy, limited resources, processing capacity as well as operating costs. Therefore, we propose an effective solution for comprehensive protection of IoT systems against attacks to network security with improved security protocols. By combining security solutions on the layers of the IoT and code improvement, algorithmic enhancement positively contributes to this important task. In the article, we propose to improve and combine the DTLS protocol and the overhearing mechanism and then have some experiments to prove the effectiveness, feasibility, economical and appropriate on popular IoT network models. The study has 5 chapters: Section 1: Introduction; Section 2: Comprehensive security solution combine improved DTLS protocol and changed overhearing mechanism; Section 3: Simulation of comprehensive security solution; Section 4: Result of simulation experiments and Section 5: Conclusion and future development.

2. COMPREHENSIVE SECURITY SOLUTION COMBINING IMPROVED DTLS AND OVERHEARING

2.1. Introduction to DTLS protocol and overhearing mechanism

Datagram transport layer security protocol (DTLS) was established by netscape communications to prevent attacks of data sniffing and spoofing at wireless sensor network (WSN) which uses user datagram protocol (UDP) [1]. The DTLS establishes the channel with asymmetric encryption in sessions layer and symmetric encryption in presentation layer so it can benefit the advantage of asymmetric encryption such as high level of safety as well as symmetric encryption such as short time for encrypting and decrypting. In the other hands, DTLS has some differences from TLS to adapt to WSN's environment. Firstly, the DTLS rises the reliability of encryption mechanisms by installing the time counter in server and client to resend the lost packets. Secondly, the length of encryption keys in DTLS is shorter than this in TLS to decrease the complexity of algorithm in WSN environment. Finally, although the DTLS is still in developing process, it is validated to be effective for preventing the attacks of data sniffing and spoofing and securing the confidentiality and integrity of WSN's data.

Meanwhile, overhearing mechanism was proposed to detect and prevent denial of service (DoS) Attack which destroys the Availability of IoT. Specifically, DoS attackdestroys availability for providing services to legal users of IoT and thus, all service operations of IoT might be forced to delay and completely stop so IoT users and provider must suffer damage about finance and reputation. There are many ways to classify DoS Attacks, but the popular classification is classifying them by types into 2 kinds: architecture and mechanism [2]. Architecture is the abstract model while mechanism is the physical way of DoS Attack in IoT. Nowadays, the most popular DoS Attack is the attack using botnet architecture and UDP flood mechanism. Overhearing will allow nodes can monitor mutually and detect Bot by searching the singularity point about IoT transmission. When the bots is detected, they will be isolated before beginning their attack.

2.2. Comprehensive security solution on IoT with improved DTLS and overhearing

Three basic characteristics of security and information safe are defined in CIA security triangle [3] concluded: integrity, availability, confidentiality. Moreover, the advance Security 6-pointed star CIA [4] is indicated in Figure 1 with the addition of more 3 advance characteristics. Each advance characteristic is the interference of two basic characteristics next to it. In Figure 1, three white peaks with upper-case-characters labels represent to the basic characteristics while three black ones lower-case-characters labels represent to the extend characteristics.

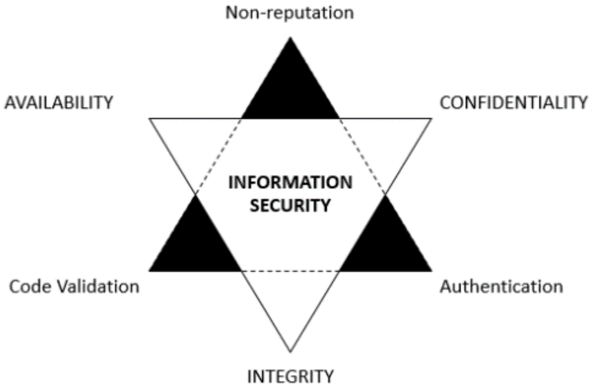


Figure 1. Indication of advance security 6-pointed star CIA

The IoT system will be safe if its security solution protects both 3 basic characteristics in the security CIA Triangle because the safety of 3 basic characteristics will ensure the safety of 3 other extend characteristics and it means all necessary characteristics of IoT System are in protection. Therefore, the combination of the integrity and confidentiality protection by DTLS protocol and availability protection by overhearing mechanism will ensure all basic characteristics are secured. In addition, the DTLS protocol and overhearing mechanism also have proven about their efficiency for preventing all data sniffing and spoofing attacks and DoS attacks using botnet and UDP flood respectively, so the deployment of both two above solutions will protect IoT Systems against almost potential risk in current world, especially IoT systems are vulnerable in DoS attack.

In the other hand, Yashaswini proposed in 2017 the comprehensive security architecture concluded 4 components such as application layer, support layer, transmission layer and sensor layer [5]. The DTLS protocol is installed in transmission layer while the overhearing mechanism is installed in sensor layer. Figure 2 will indicate the location of 2 solutions in the full IoT network.

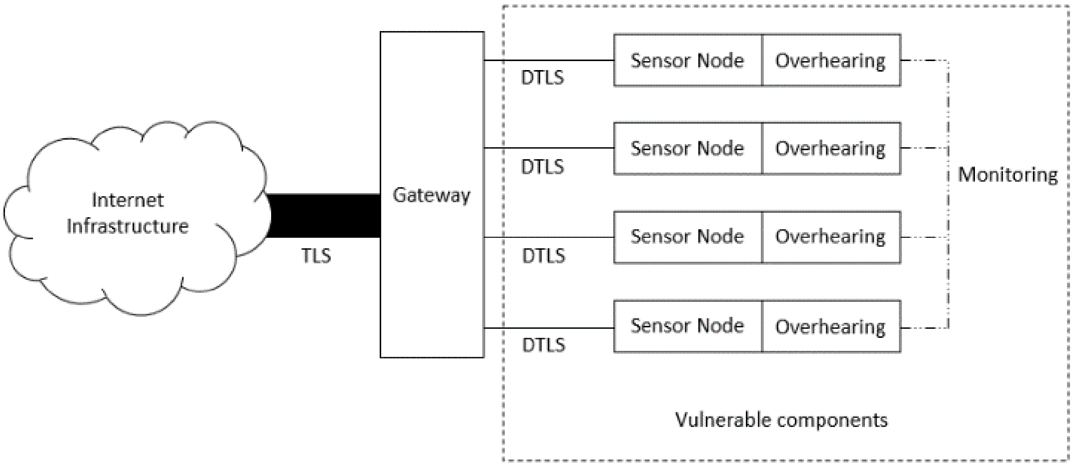


Figure 2. Location of overhearing and DTLS in IoT system

From Figure 2, both overhearing and DTLS are installed in all vulnerable components conclude sensor node, sensor environment and gateway. Firstly, these components have weak bandwidth, limited resource, and weak back-up mechanism so the DoS attack destroys the IoT network easier than the internet. Secondly, these components use IoT standards such as 6LoWPAN and Zigbee [6] also lack effective security solutions in current world while the internet Security is protected by strong security mechanisms such as TLS. Therefore, these IoT components absolutely need to be installed the comprehensive security solution.

2.3. Chalenges of combination between DTLS and overhearing in IoT system

The combination of DTLS and overhearing is not simple as popular view that they are only installed and been operating independently. Some challenges of the combination are shown in the below list:

- **Resource consumption:** DTLS mechanisms and protocol protocols must consume resource (energy and memory) in operating. However, due to limited resource of IoT, the operation of many advanced security solutions will dominate resource of other IoT components, thus can cause the IoT activities are delayed or seriously, stopped. Especially, the DTLS consume as much resource as its need to be improved to reduce consumption. Ironically, the stagnation of IoT from resource consumption of these security solutions make the overhearing deployment meaningless when its main task is maintaining the stable of IoT operation.
- **Confliction:** Both DTLS protocol and overhearing protocol also changes data and information in IoT operation, especially, the operation of DTLS encryption convert content of all packets to secure as well as make some difficulty for the monitoring process of overhearing. Nevertheless, the different location between 2 solutions decline the ability of their mutual confliction during installing and operating process.

Because of the above challengers, the installing and operating of the comprehensive security solution needs to be in careful process to research logical theory as well as experiments in simulated IoT.

3. SIMULATION OF COMPREHENSIVE SECURITY SOLUTION

3.1. Installing simulation comprehensive security solution in contiki operating system

Contiki operating system has some advantages such as open-source code providing flexibility, the accurate visualization of simulation and friendly interface [7]. Moreover, the overhearing mechanism is installed and validated in contiki operating system, so it ensures the reliability and safety to continue to the simulation in this paper. On the other hand, the DTLS also has version activated in contiki environment called "*tiny-dtls*"[8]. All simulations are designed and deployed in file via Cooja Java-code simulation software [9]. In [10], [11], our team researched and proposed the overhearing including detection of bots by "Singularity point from median Algorithm" and prevention of the DoS attack by isolating bots. We deployed them in contiki operating system as well as simulated a DoS attack by UDP Flood and botnet in square grid

WSN. The result of simulation proves the efficiency of the overhearing when the network still maintained its activities, despite short decrease of performance. In contiki operating system, overhearing mechanism is installed in source code controlling network nodes, in file framer-802154.c, folder "*core/net/mac*". Therefore, all nodes based on 2 network Standard IEEE 802.15.4 and IEEE 802.15.4e are under protection of overhearing mechanism. It is noted that the overhearing cannot affect to the default smart-consumption-saving mechanism of IoT node, and it still turns on or turns off for reducing resource consumption as the default schedule.

The simulation of DTLS protocol in contiki operating system is "*tiny-dtls*". However, the protection zone of "*tiny-dtls*" is not all simulations of contiki operation, instead of including simulations in reference links from file "project-conf.h" that "*tiny-dtls*" was declared [12]. The Table 1 following is a comparison between the DTLS protocol and the Overhearing mechanism to better understand how it works, helping us to have a more appropriate integration plan into the same system.

Table 1. The different between DTLS Protocol and overhearing mechanism

	DTLS	OVERHEARING
Location in IoT	Transmission Layer (related to Protocol)	Sensor Layer (related to WSN)
Target of protection	Confidentiality: prevent sniffing Attacks	Availability: prevent DoS Attacks
Characteristic	Integrity: prevent spoofing Attacks Symmetric and Asymmetric encryption Hash mechanism for data Data Back-up Mechanism by ECC	Each node monitors its neighbors Bots are isolated after detected by "Singularity point from median Algorithm"
Location in "contiki-master"	Folder "tiny-dtls" on folder "apps"	In file framer-802154.c of "core/net/mac"
Area of protection	Simulation has references to folder "app"	Simulation has node using Standard IEEE 802.15.4

DTLS has many complicated encryption mechanisms which consume a large amount of resource, so it is extremely difficult to operate in normal IoT network, even it does not share resource with overhearing. In fact, the resource of IoT notes is limited, like Tmote sky in MSP430 platform has only 10KB RAM memory and 48KB Flash memory. Therefore, some changes in DTLS will be deployed to ensure it declines resource consumption but maintains the good protection and stability of IoT system [13]. All changes are listed in the below list:

- **Decrease key length of advanced encryption standard (AES) encryption:** The AES encryption uses the complexity of algorithm to ensure the secure of information [14]. In the other hand, the decrease of key length of AES encryption would not decrease the complexity of algorithm but decrease the processing time and thus, decline latency of IoT. In "*tiny-dtls*", the key length is **16 bits** but it will decrease to only **8 bits**.
- **Decrease key length of secure hash algorithm (SHA):** The shorter key length the SHA is, the shorter process time is and thus, the latency decreases[15]. It is noted that the decrease of SHA key length also leads to increase probability of cypher text duplication, thus decrease the strength of SHA against the spoofing attacks but in this simulation, the sample size of appeared characters is not huge enough to cause the cypher text duplication. In "*tiny-dtls*", the key length is **32 bits** but it will decrease to only **16 bits**.
- **Eliminate the DoS countermeasures of DTLS:** In DTLS protocol, the DoS countermeasures prevents the DoS attack by providing stateless cookies in HelloVerifyRequest message from Server and ClientHello message from Client to validate whether this message was spam, thus detect DoS attacks. It is clearly that the DoS Countermeasures is not necessary because of the appearance of overhearing mechanism. Especially, the unnecessary DoS countermeasures also consume memory by increasing the allocated length of messages and cause the traffic jam if the large amount of nodes launch the "hand-shake procedure" at the same time [16]. Therefore, we eliminate the cookies of the DoS countermeasures.

DTLS, still ensures the secure of information, has enough resource to operate the Overhearing in comprehensive solutions.

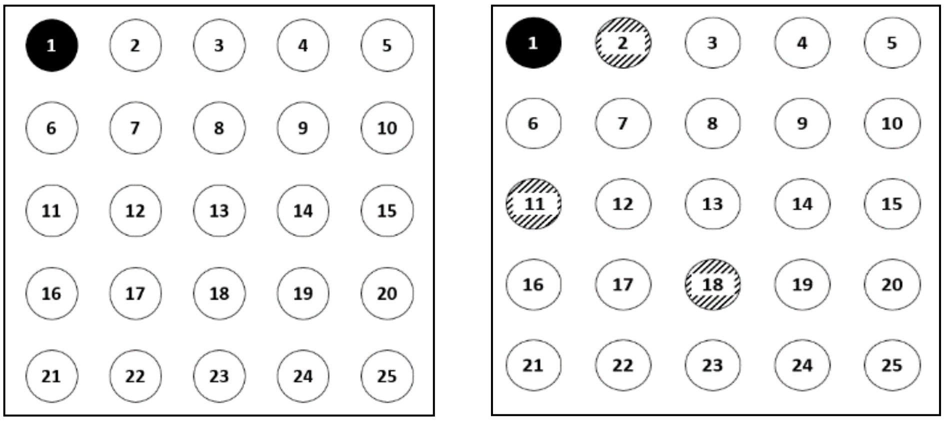
3.2. Installing simulation comprehensive security solution in contiki operating system

As the abovementioned, the target of this study is proving the reliability, the effectiveness, and the optimization in combination between DTLS and overhearing in a IoT System to design comprehensive security structure for IoT, thus, we simulate our combining solution in Contiki-OS. "tiny-dtls" is adequate to almost IoT simulations in Contiki-OS. It is necessary to validate whether the DTLS protocol prevented the overhearing activities in suffering DoS Attack by Botnet and UDP Flood, although its encryption is independent with the overhearing. We designed 6 simulation test cases with 3 of them run in normal transmission and the rest runs in overload transmission for simulating DoS Attack:

- **Test case 1 (TC 1)**: Normal transmission, no Overhearing, no DTLS.
- **Test case 2 (TC 2)**: Normal transmission, installing Overhearing, no DTLS.
- **Test case 3 (TC 3)**: Normal transmission, installing Overhearing, installing DTLS.
- **Test case 4 (TC 4)**: Overload transmission, no Overhearing, no DTLS.
- **Test case 5 (TC 5)**: Overload transmission, installing Overhearing, no DTLS.
- **Test case 6 (TC 6)**: Overload transmission, installing Overhearing, installing DTLS.

From 6TC, it is easy to comparing test cases mutually, simulation time is 5 minutes each test case.

The IoT Network uses the square grid topology which the distance between each side-by-side node is 30 meters. Therefore, according to the Pythagorean Theorem about right triangle [17], the distance between two diagonal-side nodes is approximately 42 meters. A square side concludes 5 nodes so the number of nodes in the simulation is 25 nodes. The grid topology increases the number of neighbors of each node and thus raising the flexibility in routing. In this simulation, the transmission range is 50 meters so with this topology, a node can operate direct transmission to all side-by-side nodes in horizontal, vertical, and diagonal directions. Thus, a node can transmit directly to 3 nodes if it locates in the corner, 5 nodes if it locates in the side and 8 nodes if it locates in the middle of square topology. It is noted that the multi-hop which allow a node to transmit to more than 1 destination did not exist in this simulated experiment. The Figure 3 indicates this topology:



(a) (b)

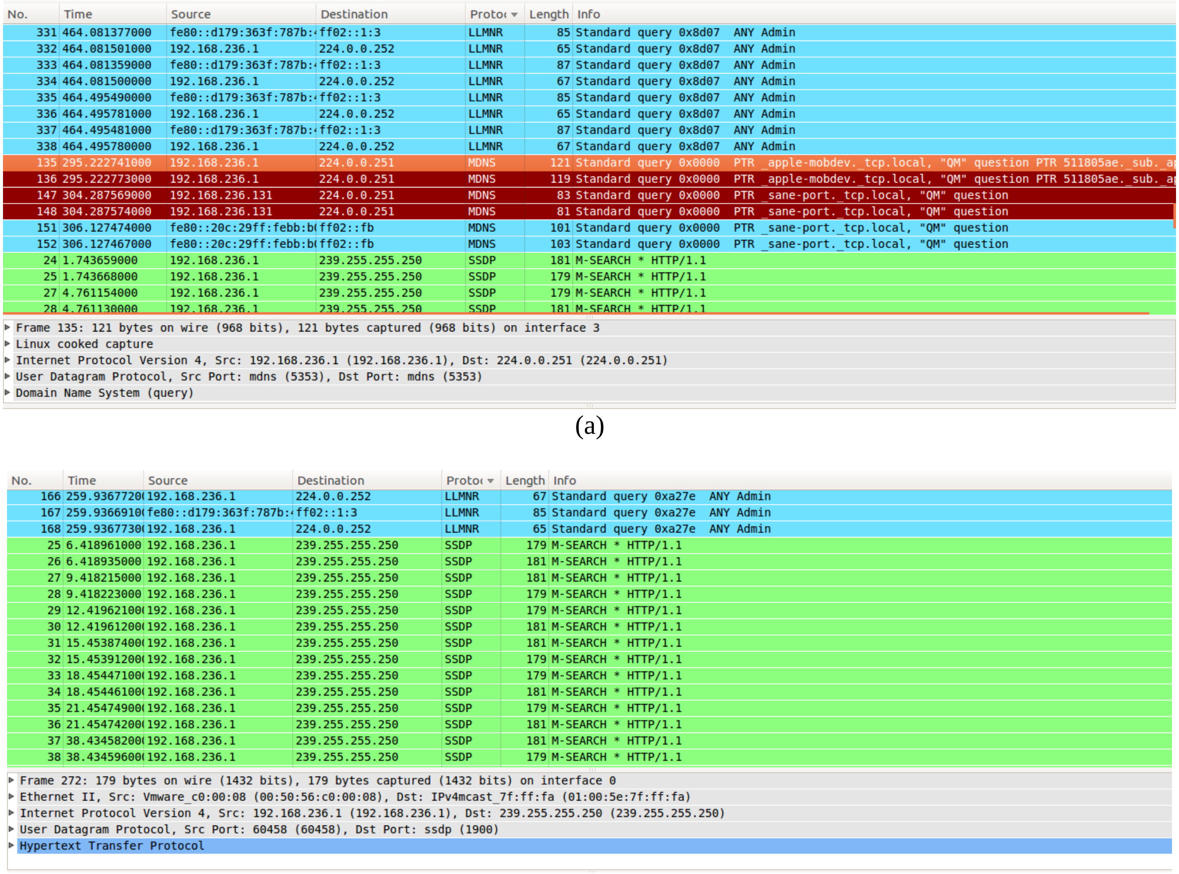
Figure 3. Topology of IoT network in simulation test cases, (a) Nomal transmission test cases, (b) Overload transmission test cases

In Figure 3(a) and (b), nodes with black background color and white character color are Server nodes while nodes with white background color and black character color are client nodes. In Figure 3(b), there are 3 Bot nodes with dark-upward-diagonal background pattern which launch DoS attack by sending large amount of UDP Packets to dominate the resource of server. The location of Bot is various: different routing distance from server node, different position in DAG tree and this diversity ensure the DoS attack effects all IoT network.

4. RESULTS AND DISCUSSION

4.1. Decapsulate and analyze packets

Wireshark was selected to analyze and evaluate our proposal because it is the useful tools to research and gather the network protocol based on data from decapsulated packets for evaluating the protocol [18]. The test case is run with wireshark is Test case TC3 representing to network installing DTLS and Test case TC2 representing to network without installing DTLS. From Figure 4, the appearance of MDNS [19] packets in network installing DTLS (in dark brown label) proved the operation of DTLS Protocol.



```
No.  Time          Source                Destination      Protoc  Length Info
 331 464.081377000 fe80::d179:363f:787b: ff02::1:3        LLMNR       85 Standard query 0x8d07  ANY Admin
 332 464.081501000 192.168.236.1         224.0.0.252      LLMNR       65 Standard query 0x8d07  ANY Admin
 333 464.081359000 fe80::d179:363f:787b: ff02::1:3        LLMNR       87 Standard query 0x8d07  ANY Admin
 334 464.081500000 192.168.236.1         224.0.0.252      LLMNR       67 Standard query 0x8d07  ANY Admin
 335 464.495490000 fe80::d179:363f:787b: ff02::1:3        LLMNR       85 Standard query 0x8d07  ANY Admin
 336 464.495781000 192.168.236.1         224.0.0.252      LLMNR       65 Standard query 0x8d07  ANY Admin
 337 464.495481000 fe80::d179:363f:787b: ff02::1:3        LLMNR       87 Standard query 0x8d07  ANY Admin
 338 464.495780000 192.168.236.1         224.0.0.252      LLMNR       67 Standard query 0x8d07  ANY Admin
 135 295.222741000 192.168.236.1         224.0.0.251      MDNS       121 Standard query 0x0000  PTR _apple-mobdev._tcp.local, "QM" question PTR 511805ae._sub._ap
 136 295.222773000 192.168.236.1         224.0.0.251      MDNS       119 Standard query 0x0000  PTR _apple-mobdev._tcp.local, "QM" question PTR 511805ae._sub._ap
 147 304.287569000 192.168.236.131       224.0.0.251      MDNS        83 Standard query 0x0000  PTR _sane-port._tcp.local, "QM" question
 148 304.287574000 192.168.236.131       224.0.0.251      MDNS        81 Standard query 0x0000  PTR _sane-port._tcp.local, "QM" question
 151 306.127474000 fe80::20c:29ff:febb:b ff02::fb         MDNS       101 Standard query 0x0000  PTR _sane-port._tcp.local, "QM" question
 152 306.127467000 fe80::20c:29ff:febb:b ff02::fb         MDNS       103 Standard query 0x0000  PTR _sane-port._tcp.local, "QM" question
  24 1.743659000   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  25 1.743668000   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  27 4.761154000   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  28 4.761130000   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
▸ Frame 135: 121 bytes on wire (968 bits), 121 bytes captured (968 bits) on interface 3
▸ Linux cooked capture
▸ Internet Protocol Version 4, Src: 192.168.236.1 (192.168.236.1), Dst: 224.0.0.251 (224.0.0.251)
▸ User Datagram Protocol, Src Port: mdns (5353), Dst Port: mdns (5353)
▸ Domain Name System (query)
```

(a)

```
No.  Time          Source                Destination      Protoc  Length Info
 166 259.9367720   192.168.236.1         224.0.0.252      LLMNR       67 Standard query 0xa27e  ANY Admin
 167 259.9366910   fe80::d179:363f:787b: ff02::1:3        LLMNR       85 Standard query 0xa27e  ANY Admin
 168 259.9367730   192.168.236.1         224.0.0.252      LLMNR       65 Standard query 0xa27e  ANY Admin
  25 6.418961000   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  26 6.418935000   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  27 9.418215000   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  28 9.418223000   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  29 12.41962100   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  30 12.41961200   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  31 15.45387400   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  32 15.45391200   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  33 18.45447100   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  34 18.45446100   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  35 21.45474900   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
  36 21.45474200   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  37 38.43458200   192.168.236.1         239.255.255.250  SSDP       181 M-SEARCH * HTTP/1.1
  38 38.43459600   192.168.236.1         239.255.255.250  SSDP       179 M-SEARCH * HTTP/1.1
▸ Frame 272: 179 bytes on wire (1432 bits), 179 bytes captured (1432 bits) on interface 0
▸ Ethernet II, Src: Vmware_c0:00:08 (00:50:56:c0:00:08), Dst: IPv4mcast_7f:ff:fa (01:00:5e:7f:ff:fa)
▸ Internet Protocol Version 4, Src: 192.168.236.1 (192.168.236.1), Dst: 239.255.255.250 (239.255.255.250)
▸ User Datagram Protocol, Src Port: 60458 (60458), Dst Port: ssdp (1900)
▸ Hypertext Transfer Protocol
```

(b)

Figure 4. The appearane of MDNS messages in transmission of network installing DTLS, (a) Network installing DTLS, (b) Network without installing DTLS

4.2. Decapsulate and analyze packets

We have 3 criterions to measure the WSN performance:

4.2.1. Packets delivery rate (PDR)

PDR is rate between the number of received packets and the number of sent packets. The unit of PDR is percent (%). In (1) calculates PDR:

$$PDR = \frac{R}{S}\, x\ 100 \quad . \tag{1}$$

In (1), S is the number of packets the calculating node sent while R is the number of packets the other nodes received from calculating node. PDR represents the reliability of transmission, the higher PDR is, the higher successful rate of transmission is [20]. According to Mansfield from Cengage Center, PDR from higher 95% to ensure the WSN has stable operation [21].

4.2.2. Latency

Latency is the average time a packet between departing from sender (calculating node) and arriving to receiver. The basic unit of Latency is milliseconds (ms). In (2) calculates Latency:

$$Latency = \frac{\sum_{i=1}^{n}(t(R)_i - t(S)_i)}{n} \tag{2}$$

In (2), n is number of successful transmission packets, i is the index of packet, T(S)i is the time the calculating node sent packet index i while T(R)i is the time the receiver received packet index i. The Latency represents the quality of transmission, the higher the latency is, the longer time for transmitting is [18].

According to SAS Information Technology Service Center in the United Kingdom, the Latency must be lower than 800 ms to ensure the WSN has stable operation [22].

4.2.3. Energy consumption:

Energy consumption is the abstract criterion represents to which amount of energy is consumed in different simulation activities. In contiki, the energy consumption is calculated by the rate between the time node for different tasks (sending packets, receiving packets) and total time of simulation. However, Sourceforge proposed the (3) to calculate energy consumption measured by milli Joule (mJ) from the abstract value [13]:

$$E = (Tx\ x\ 19.5 + Rx\ x\ 21.8 + CPU\ x\ 1.8 + LPM\ x\ 0.545)\ x\ \frac{3}{32768} \quad (3)$$

In (3), Tx is the rate between time a node uses to send packets and total simulation time while Rx is the rate between time a node uses to receive packets and total simulation time. CPU is energy consumption of CPU for simulation (different kind of node has different CPU value) and LPM is the rate between the time a node uses for basic tasks of node and total simulation time. In total WSN, we will measure three criterions and take the average value of all nodes in WSN. The kind of node in simulation is Tome Sky which require energy consumption lower than 12.6 mow [23] in an hour, 1.05 mew in 5-minutes simulation. It means the energy consumption must be lower than 315 mJ to ensure the WSN has stable operation. The Table 2 indicates result of the experiment result from TC1 to TC6. It is noted that the value of this results is the average value of IoT with each criterion.

Table 2. Performance of WSN test cases

	Condition and installed mechanism	PDR (%)	Latency (ms)	Energy (mJ)
TC1	Normal, DTLS	98.54	617.15	142.19
TC2	Normal, Overhearing	98.51	632.34	167.6
TC3	Normal, DTLS, Overhearing	95.28	645.24	201.56
TC4	Overload, DTLS	16.59	50912.11	991.73
TC5	Overload, Overhearing	96.9	714.27	195.13
TC6	Overload, DTLS, Overhearing	95.03	793.64	302.7

From Table 2, there is some analysis statements: In normal transmission, the deployment of comprehensive security solution combining DTLS and overhearing decreased the performance of WSN, especially, the energy consumption increased a large amount in a small period. The reason of this decrease is the operation of both DTLS and overhearing also consume resource of WSN. For example, the encryption mechanisms of DTLS and the monitoring mechanisms of overhearing also increased energy consumption of WSN. However, the decrease completely did not delay the operation of WSN, the PDR and Latency was still above threshold for stable transmission. The rise of energy consumption did not make the WSN exhausted [24].

In overload transmission from a simulated DoS attack, the overhearing detected this attack early restricted its consequence. All criterions of WSN despite decreasing but still above threshold for stable transmission, even the DTLS consumed a large amount of energy. Soon, we will deploy the comprehensive security solution combining DTLS protocol and overhearing mechanism in real IoT system with arduino devices [25].

In conclusion, the experiment simulating comprehensive security solution completes all its tasks. Firstly, the operation, reliability, and efficiency of both DTLS and overhearing was validated by capturing packets as well as monitoring WSN performance in overload transmission caused by a DoS attack. Secondly, the unavoidable challenges of this comprehensive security solution appeared by the decrease of all performance criterions of WSN installing DTLS and overhearing, although this decrease was not large (lower than WSN without installing the solutions 10%) and still above threshold for stable transmission in current WSN.

5. CONCLUSION AND FUTURE DEVELOPMENT

From this paper, we classified all risks at Security and Information Safety in IoT System by 3 information security features need to be protected via CIA security triangle including confidentiality (against sniffing attack), Integrity (against spoofing attack) and availability (against DoS attack). Our team proposed independent security solutions such as DTLS Protocol preventing sniffing and spoofing attack and overhearing mechanism preventing DoS Attack. The aim of this paper, our team proposed a comprehensive security solution combining improved DTLS in transmission layer and sensor layer. Our solution includes location diagram, improvements to decrease resource consumption of DTLS for adapting to low-energy network. After proposed theoretical basis, our team sets up simulating experiments in contiki operating system to deploy comprehensive security combining improved DTLS and overhearing as well as simulate a DoS Attack by Botnet and UDP Flood. Our team used Wireshark to capture packet and proved the appearance of DTLS encryption in simulated WSN transmission. Meanwhile, the results about WSN performance such as PDR, Latency proved the stable operation of WSN with installing this comprehensive security solution and suffering a simulated DoS Attack. The deployment of real IoT System has more challenges than this in simulated IoT system but the reliability and the reliability of experiment in real system is much higher than in the simulated system in contiki operating system. This comprehensive security solution was researched hardly in long period of time as well as based on previous study published in past conferences, the theoretical basis and simulated experiment could not avoid mistakes, so our team completely want to receive the advice as well as cooperate from scientists in areas of IoT security.

ACKNOWLEDGMENT

Our team sends the gratefulness to tutors and the colleagues for providing adequate support about infrastructure and advice during the researching process. They worked in Bach Khoa Cipher Security Center and other faculties in School of Information Communication Technology, Hanoi University of Science and Technology as well as Vietnam National University. Supported by Cortek Co., ltd in KC01.15 project.

REFERENCES

[1] T. Dierks, and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246,” *Proposed Standard, Internet Engineering Task Force*, vol. 1, no. 2, p. 5246, 2008.

[2] L. Kaur Somal, and Karanpreet Singh Virk, “Classification of Distributed Denial of Service Attacks – Architecture, Taxonomy and Tools,” *Computer Science and Technology*, vol. 2, no. 2, pp. 118-122, 2014.

[3] S. Samonas, and D. Coss, “The CIA Strikes Back: Redefining Confidentiality, Integrity and Availability in Security,” *Journal of Information System Security*, vol. 10, no. 3, pp. 21-45, 2014.

[4] A. Pratt, “CIA Triad and New Emerging Technologies: Big Data and IoT,” *Los Angeles City College and Consultant*, 2015.

[5] J. Yashaswini, “A Review on IoT Security Issues and Countermeasures,” *Orient.J. Computer Science and Technology*, 2017, doi: 10.13005/ojcst/10.02.28.

[6] D. Xu Li, Wu He, and Shancang Li, “Internet of things in industries: A survey,” *IEEE Transactions on industrial informatics*, vol. 10, no. 4, pp. 2233-2243, 2014, doi: 10.1109/TII.2014.2300753.

[7] A. Dunkels, “Contiki: Bringing IP to Sensor Networks,” *ERCIM News Journal, European Union*, 2009.

[8] O. Bergmann, Simon Bernard, and Julien Vermillard, “Tiny-dtls – Eclipse IoT,” *Eclipse Foundation*, 2011.

[9] Pedro, Mugdhe, and Samarth, “Cooja Simulator – Contiki Tutorials,” *Autonomous Networks Research Group*, University of South California, 2016.

[10] T. Nguyen Van, Tri Ngo Quang, Tuyen Nguyen Gia, Duc Tran Quang, Anh Tran Hai, and Tung Bui Trong, “The Flooding Attack in Low Power and Lossy Networks: A Case Study,” *The 7 th IEEE International Conference on Smart Communications in Network Technologies*, El Oued, Algeria, 2018, pp. 183-187, doi: 10.1109/SaCoNeT.2018.8585451.

[11] T. Nguyen Van, Tri Ngo Quang, Tuyen Nguyen Gia, Giang Linh Nguyen, and Tien Nguyen Viet, “Design a Security System for Internet of Things with detectinng and eliminating Denial of Service Attack based on Overhearing mechanism,” the *3rd Symposium of Information Security*, Vietnam, 2018.

[12] A. Velinov, and A. Mileva, “Running and Testing Applications for Contiksi OS Using Cooja Simulator,” *International Conference on Information Technology and Development of Education*, 2016, pp. 279-285.

[13] M. Abdellatif, “[Contiki Developer] Power Consumption,” 2017.

[14] J. Daemen, and Vincent Rijmen, “The Design of Rijndael: AES - The Advanced Encryption Standard,” *Springer-Verlag*, 2002, doi: 10.1007/978-3-662-04722-4.

[15] S. Marc, Bursztein Elie, Karpman Pierre, Albertini Ange, and Markov Yarik, “The first collision for full SHA-1,” *Google Research*, 2017.

[16] E. Rescorla, and N. Modadugu, “Datagram Transport Layer Security Version 1.2,” *Internet Engineering Task Force*, vol. 1, no. 2, p. 15, 2012.

[17] W. Sierpinski, “Pythagorean triangles,” *New York: Dover Publications*, vol. 9, 2003.

[18] G. Combs, “Q&A with the founder of Wireshark and Ethereal,” *Interview in protocolTesting.com*, 2016.

[19] S. Cheshire, and M. Krochmal, “Multicast DNS,” *Internet Engineering Task Force (IETF)*, 2013.

[20] D. Culler, D. Estrin, M. Srivastava, “Guest Editors’ introduction: overview of sensor networks,” *Computer, vol. 37, no. 8*,p41-49, 2004, doi: 10.1109/mc.2004.93.

[21] K. C. Mansfield, and J. L. Antonakos, “Computer Networking from LANs to WANs: Hardware, Software, and Security,” *Boston Cengage Learning*, p. 26, 2010.

[22] SAS Team, “What is network latency and how do you use a latency calculator to calculate throughput?,” *The SAS Group of Companies Limited*, 2019.

[23] M. Johnson, *et al.*, “A Comparative Review of Wireless Sensor Network Mote Technologies,” *IEEE SENSORS Conference*, 2009, p 1442, doi: 10.1109/ICSENS.2009.5398442.

[24] D. Hofstrand, “Energy measurements and conversions,” *Iowa State University Extension and Outreach*, 2007.

[25] T. Nguyen Tat Bao, and H. Pham Quang, “Arduino and IoT Programming,” *Vietnam Education Publishing House Ltd Co*.

BIOGRAPHIES OF AUTHORS

Dr. Nguyen Van Tanh graduated from Hanoi University of Science and Technology, Vietnam. Currently a lecturer and researcher at the International School – Vietnam National University, Hanoi. Author of many Research on information security, security for IoT networks and many scientific publications in the field of IT applications. Address: 144 Xuan Thuy Street, Cau Giay District, Hanoi City, Vietnam. Email: Tanhnv@vnu.edu.vn

Ngo Quang Tri. Graduated from Hanoi University of Science and Technology. As a researcher, the author publishes many scientific articles in the fields of information technology, information security and computer network communication. Currently working in the Information Technology field. Address: 01 Dai Co Viet Street, Hai Ba Trung District, Hanoi City, Vietnam. Email: Tri.ngoquang@sie.edu.vn

Mai Manh Trung graduated from, Vietnam National University, Hanoi. Currently a lecturer at the University of Economics Technology for Industries and researcher at the University of Engineering and Technology, Vietnam National University, Hanoi. Research area: Cryptography, lightweight cryptography, security for IoT networks, artificial intelligence, application programming. Address: 144 Xuan Thuy Street, Cau Giay District, Hanoi City, Vietnam. Email: mmtrung@uneti.edu.vn